

## Il computer connesso – 1

Epistemologia, Deontologia ed Etica dell'Informatica  
Storia dell'Informatica e della Comunicazione Digitale

**Federico Gobbo**

federico.gobbo@uninsubria.it

CRII – Centro di Ricerca “Informatica Interattiva”

Università dell'Insubria, Varese-Como

© Alcuni diritti riservati.

A.A. 2010-11

## Da ARPAnet a Internet

Nel **1982** nasce l'esigenza di mettere in collegamento la rete ARPAnet con le reti, e a tal scopo viene deciso lo standard di comunicazione, basato sui due protocolli **TCP** e **IP**. Per la prima volta viene usato il termine **Internet**, nel documento **RFC 675**.

Verso la fine degli anni 1980 TCP/IP collega i principali centri di ricerca in America ed Europa, e altre istituzioni rilevanti.

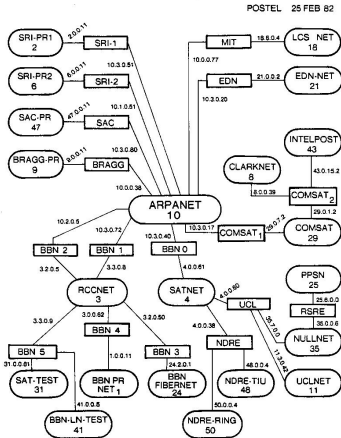


Figura: Test della rete Internet su TCP/IP (1982)

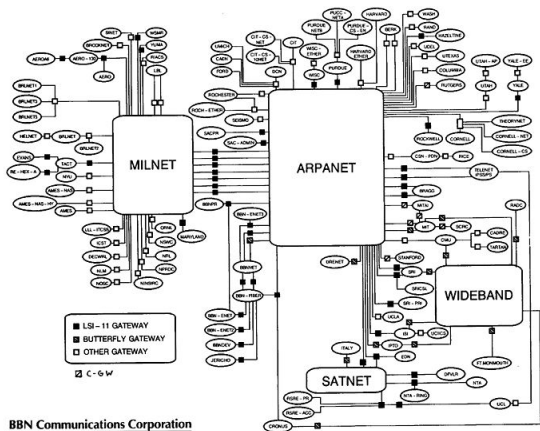


Figura: La rete Internet nel 1986

## 1991, nasce il Web

Nel **1991**, ispirandosi ai lavori di Vannevar Bush e Ted Nelson – in particolare il protocollo FEBE, e avendo lavorato su HyperCard di Apple, **Tim Berners-Lee**, fisico inglese che lavora al **CERN** di Ginevra, un importante centro di ricerca svizzero, scrive il primo **server web** della storia, con lo standard **HTML** e il protocollo **HTTP**.

Nel **1993** il CERN rinuncia a qualsiasi diritto d'autore sui nuovi standard decretandone il successo mondiale. Nello stesso anno esce il primo browser web, il **Mosaic**.



**Figura:** Il cubo con cui Tim Berners-Lee ha inventato il Web

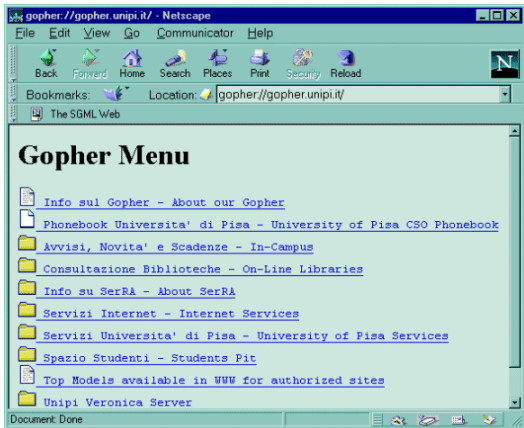


Figura: Navigazione internet con Veronica (motore Gopher)



Figura: Il primo browser web, il Mosaic (1993; Computer History Museum)



## Il progetto GNU, A.D. 1990

“Nel 1990 il sistema [operativo] GNU era quasi completo; l'unica componente maggiore che mancava era il kernel.” (Stallman, 1999). La difficoltà della FSF di rendere il kernel

GNU (HURD) stabile rende scettica la comunità degli sviluppatori. Inoltre, l'effetto virale della GPL non piace perché impedisce la combinazione con il software proprietario, e in ultima analisi rende difficile venderlo.

Stallman risponde: regalate il codice, vendete il servizio di assistenza.

# 1991, nasce Linux

```
> From: torvalds@klaava.Helsinki.FI (Linus Benedict Torvalds)
> Newsgroups: comp.os.minix
> Subject: What would you like to see most in minix?
> Summary: small poll for my new operating system
> Message-ID: <1991Aug25.205708.9541@klaava.Helsinki.FI>
> Date: 25 Aug 91 20:57:08 GMT
> Organization: University of Helsinki
>
> Hello everybody out there using minix -
>
> I'm doing a (free) operating system (just a hobby, won't be big and
> professional like gnu) for 386(486) AT clones. This has been brewing
> since april, and is starting to get ready. I'd like any feedback on
> things people like/dislike in minix, as my OS resembles it somewhat
> (same physical layout of the file-system (due to practical reasons)
> among other things).
>
> I've currently ported bash(1.08) and gcc(1.40), and things seem to work.
> This implies that I'll get something practical within a few months, and
> I'd like to know what features most people would want. Any suggestions
> are welcome, but I won't promise I'll implement them :-)
```

>

```
> Linus (torvalds@kruuna.helsinki.fi)
>
> PS. Yes - it's free of any minix code, and it has a multi-threaded fs.
> It is NOT protable (uses 386 task switching etc), and it probably never
> will support anything other than AT-harddisks, as that's all I have :-).
```

# Linus Torvalds nel film *The Code*, 21 anni



## Linux-Minix: 1-0 per Torvalds!

Linus posta il suo lavoro sul kernel nel newsgroup del microkernel Minix (il web, per quanto nato da un anno circa, non era diffuso perché mancava un browser decente).

Il microkernel Minix era (a) sperimentale; (b) piú complesso rispetto a un kernel monolitico; (c) lento. Linux invece nasce come kernel monolitico. Tanenbaum scrive a Linus:

“Linux è obsoleto... Progettare un kernel monolitico nel 1991 è un errore di fondo.” Linus, ventunenne, risponde: “il suo lavoro è fare il ricercatore e il professore: è una dannata scusa per i danni cerebrali di Minix.”

## Linux o GNU/Linux? Stallman vs. Torvalds

Fino al kernel Linux 0.11+VM (fine 1991) la licenza di Linus era di questo tipo: “- It’s free (copyright by me, but freely distributable under a very lenient copyright)”.

Dalla versione 0.12 (5 gennaio 1992), Linux passa alla GPL. Perché? “Perché il compilatore GCC è sotto GPL.”

Stallman allora minaccia di fare causa a chiunque usi Linux in combinazione con software non-GPL. La risposta di Torvalds è sorprendente e creativa.

## Linus interpreta creativamente la GPL (1999, T.d.A.)

*Abbiamo finito per decidere (o forse ho finito io per decretare) che le chiamate di sistema non sarebbero state considerate collegate al kernel. In altri termini, qualsiasi programma che giri sopra Linux non verrà considerato protetto da GPL. Questa decisione fu fatta molto presto e ho addirittura aggiunto un file speciale leggimi (vedi Appendice B) per essere sicuri che tutti potessero saperlo. Grazie a questo le aziende possono scrivere programmi per Linux senza preoccuparsi della GPL. Il potere di Linux è tanto nella comunità di cooperazione che ci sta dietro quanto nel codice stesso. Se Linux venisse rapito – se qualcuno tentasse di farne e una versione proprietaria e distribuirlo – tutta la sua appetibilità, che è essenzialmente il modello di sviluppo dell'open source, sarebbe perso per la versione proprietaria.*

## Goodbye, “free software”; hello, “open source”

Eric Steven Raymond è un hacker della generazione di Stallman. Autore di Fetchmail, uno dei primi contributori al progetto GNU, ha scritto il dizionario del gergo degli hacker. Nella primavera del 1993 scopre Linux, e ne rimane folgorato. Capisce che c'è qualcosa di nuovo.

Il 3 febbraio 1998 a Palo Alto, California, Raymond inventa l'espressione “open source”, in risposta all'annuncio dato dalla Netscape di rilasciare il codice sorgente libero in rete per il loro browser, avendo perso la “guerra dei browser” contro Microsoft Internet Explorer, che dominerà il mercato incontrastato.

L'intento era evitare che il motore (Gecko) di Netscape fosse posto sotto licenza GPL: “dobbiamo far accettare il software libero nel *mainstream corporate world*” (Raymond, 1998).

## *Open source, versione 1.0*

“Open Source” non significa solo accesso al codice sorgente. Una licenza deve avere diversi requisiti, tra cui:

1. libertà di redistribuzione. Identica alla libertà 2 della GPL.
2. accesso al codice sorgente.
3. diritto (e non dovere, come nella GPL!) di usare la stessa licenza per i lavori derivati.
4. integrità del codice sorgente dell'autore, i.e. l'autore può chiedere che un lavoro derivato abbia un nome diverso (principio dei fork). Questo implica l'attribuzione del codice a un autore, cosa negata invece dalla GPL – non a caso il modello anonimo del wiki usa una licenza di questo tipo, la FDL.



## Open vs. free

Insieme a Bruce Perens, autore del Contratto Sociale Debian (e coautore di *Toy Story*), fonda la *Open Source Initiative* (OSI), che non scrive licenze ma certifica quali seguono i requisiti *open source* e quali no.

Ben presto OSI entra in un conflitto di interessi con la FSF di Stallman.

## La cattedrale come contrapposizione del bazaar

*[Il software] viene attentamente lavorato a mano da singoli o da bande di maghi che lavorano in silenzio, senza che alcuna versione beta veda la luce prima del momento giusto... [come in una] cattedrale, da costruire in silenzio e reverenza (Raymond 1997, T.d.A.)*

Il sapere perpetrato dalla cattedrale è chiuso esoterico; viceversa, il sapere del bazaar è aperto, essoterico, al di là dell'apertura del codice sorgente.

## Il progetto GNU (ma non solo) come cattedrale

*I programmi GNU sembravano “cattedrali”, monumenti all’etica hacker, impressionanti, pianificati in modo centralizzato, costruiti per durare nel tempo. [...] Linux (anche se era solo la punta dell’iceberg) divenne così la prima vera e concreta dimostrazione al mondo intero che la comunità hacker e il movimento per il software libero non era solo uno scoordinato gruppo di visionari idealisti e che le cose potevano realmente (e drasticamente cambiare). (Williams in Aliprandi 2005:21)*

## Il bazaar di Raymond come mercato dell'abbondanza

*[Il modello del bazaar] è un pullulare di progetti e approcci differenti basati sulla rapidità dei rilasci... il mercato interno della reputazione della comunità esercita una sottile pressione sulle persone in modo che non si lancino dei progetti se non si è abbastanza competenti per seguirli (Raymond 1997, T.d.A.).*

## Linux come prototipo del modello a bazaar

*Lo stile di sviluppo di Linus Torvalds – rilascia spesso e volentieri, delega tutto quello che puoi, sii aperto fino alla promiscuità – arrivò come una sorpresa. Nessuna costruzione a cattedrale, silenziosa e riverente – invece, la comunità Linux sembrava assomigliare a un vociante bazaar di approcci e priorità diverse (simboleggiate dai siti di archivio di Linux, che accettano proposte da chiunque), fuori dal quale sembrava potesse emergere un sistema coerente e stabile solo mediante una successione di miracoli. Il fatto che questo stile a bazaar sembrò funzionare, e funzionare bene, fu per me un vero shock. (Raymond in Halvelik 2004: 119)*

## Il punto di vista di Raymond...

*Ci sono tre livelli, uno ideologico, uno pratico e uno personale. Su quello ideologico... la differenza è molto chiara. Richard vuole coscientemente cambiare il mondo e esplicitamente, in una crociata morale. Se non prendi le sue argomentazioni morali astratte, non sposi la sua causa. La via che Linus Torvalds e io, e altri, hanno portato avanti è pragmatica. C'è un livello pratico sul quale c'è un conflitto sulle tattiche da usare per diffondere il messaggio. C'è anche un livello personale in cui, be', le persone hanno un ego e hanno investito nelle ideologie, e se parte di ciò che pensi sia il tuo pensiero rivoluzionario hai l'impressione te l'abbiano rubato, provi del risentimento. (Raymond in Holmevik 2004:126)*

## ...e quello di Stallman

*La differenza fondamentale tra i due movimenti risiede nei loro valori, nel loro modo di guardare al mondo. Per il movimento open source, la questione se il software dovrebbe essere a sorgente aperto è pratica, non etico. Come si dice comunemente, “open source è una metodologia di sviluppo, il software libero un movimento sociale”. Per il movimento open source il software non libero è una soluzione subottimale. Per il movimento del software libero, il software non libero è un problema sociale e il software libero è la soluzione.*

Richard Stallman, *Perché 'free'* (T.d.A.)

## Linux come 'oltreunix' e 'antimicrosoft'

*Vedo Linux come qualcosa che non è Microsoft – una reazione contro Microsoft, né piú né meno. Non penso che avrà molto successo nel lungo periodo. Ho guardato il sorgente e ci sono pezzi buoni e pezzi non buoni. Un insieme disordinato di persone hanno contribuito a questo sorgente, e la qualità cambia drasticamente.*

Ken Thompson, 1999 (in Holmevik 2004:130)

Cosa ne pensate della posizione di Thompson, coautore di C e Unix?



# La selva dei sistemi operativi liberi all'anno 2000



## Microsoft ha paura di Linux: il nuovo Davide contro Golia?

1. OSS pone nel breve periodo una sfida diretta a Microsoft, ai prodotti e alla piattaforma, in particolare nell'area server. Inoltre, il parallelismo intrinseco e lo scambio libero di idee in OSS ha benefici che non sono replicabili nel nostro modello corrente di licenze e pertanto presenta una sfida nella condivisione delle idee dello sviluppo nel lungo periodo [...].
2. Linux può vincere finché i servizi / protocolli saranno cose utili.
3. Progetti OSS sono stati in grado di guadagnare terreno in molte applicazioni server grazie al vasto utilizzo di protocolli semplici e molto utilizzati. Estendendo questi protocolli e sviluppandone di nuovi, possiamo impedire a progetti OSS di entrare nel mercato.

(Raymond Halloween I in Holmevik 2004:147)

## Linux diventa un fenomeno mediatico



Figura: Copertina di *Wired*, novembre 2003

## L'entusiasmo: abbiamo cambiato il mondo

*Quando vedi delle URL sui sacchetti della spesa, sui giornali popolari, sui fianchi dei camion, alla fine di un film subito dopo il logo del produttore – quelle eravamo noi, l'abbiamo fatto noi. Abbiamo dato Internet alle gente comune. Abbiamo dato il calcio d'inizio di un nuovo mezzo di comunicazione. Abbiamo cambiato il mondo. (Zawinski in Holmevik 2004: 127)*

## Grazie. Domande?



Potete scaricare questa presentazione qui:

<http://www.slideshare.net/goberiko/>

© CC BY ND 3.0 Federico Gobbo 2010 di tutti i testi. Pubblicato in Italia.  
Attribuzione – Non commerciale – Condividi allo stesso modo 2.5

©delle figure degli aventi diritto. In caso di violazione, scrivere a:  
federico.gobbo@uninsubria.it.