

# Il Software open

- Una chiacchierata per capire come...
  - Come nasce
  - Come viene sviluppato
  - Come sono organizzati i progetti
  - Come sono distribuiti
  - Come sono finanziati
  - ...

## Come nasce un software open (1)

- “Ogni buon lavoro software inizia dalla frenesia personale di uno sviluppatore” (*I#esr*)
- *Linux* nacque più o meno così:

Do you pine for the nice days of minix-1.1, when men were men and wrote their own device drivers? Are you without a nice project and just dying to cut your teeth on a OS you can try to modify for your needs? ...

As I mentioned a month(?) ago, I'm working on a free version of a minix-lookalike for AT-386 computers. It has finally reached the stage where it's even usable... (*Linus*)

# Riferimenti

- “La cattedrale e il bazar” (Eric S. Raymond)
  - <http://www.catb.org/~esr>
  - nota come “il manifesto del movimento opensource”
  - famosa per aver “convinto” Netscape a rilasciare i sorgenti... (mai sentito Mozilla/Firefox?)
  - una sorta di decalogo, tratto dall'esperienza dell'autore
- La mia esperienza personale (è scritto in piccolo perchè è poca, ma meglio che niente...)
  - <http://www.freepops.org>

## Come nasce un software open (2)

- Uno sviluppatore ha una necessità
- Un buon sviluppatore si diverte a scrivere programmi e inizia a implementare la sua idea
- Quando ha qualcosa di usabile (più o meno funzionante) lo rende pubblico e lo rilascia sotto licenza opensource
- Se il progetto piace a qualcuno...

## Come nasce un software ~~open~~ (3)

- Non lo so, ma probabilmente da un'idea di un manager/dipendente/cliente e da un'analisi di mercato
- Project manager, team di sviluppo
- Si raggiunge una versione definitiva o quasi e la si prova a vendere
- Se vende bene, se no...

## Il *Bazaar* (2)

- Sviluppo distribuito:
  - Il software open esiste da prima dell'avvento di internet, ma la rete è cruciale e ne ha portato all'esplosione
  - Sistemi per comunicare: mail, news, forum...
  - Sistemi per “collaborare” nello sviluppo: CVS, diff...
  - Sistemi per pubblicare e dare visibilità: web...

## Come è organizzato lo sviluppo open (1) aka il *Bazaar*

- il *Bazaar* (*esr*)
  - distribuito (sparsi per... la rete)
  - caotico (non c'è un vero capo)
  - non ci sono scadenze
  - non ci sono segreti
  - c'è di tutto, si prende quello che serve in giro
  - non ci sono “veri concorrenti”

## il *Bazaar* (3)

- Caotico
  - Non c'è un vero capo, ma c'è/ci sono delle figure di riferimento per il progetto: fondatore, maintainer corrente... insomma il “coordinatore”
  - C'è la “reputazione” che si conquista sul campo
  - C'è l'inventiva personale, si è liberi di proporre le proprie soluzioni, fare le proprie modifiche e dimostrare che funzionano...

## il Bazaar (3)

- Non ci sono scadenze (e garanzie)
  - Si può rilasciare quando si vuole (non bisogna mobilitare media)
  - “Distribuisci presto. Distribuisci spesso. E presta ascolto agli utenti” (7#esr)
- Non ci sono segreti
  - Tutto il prodotto è disponibile, con un po' di buona volontà lo si può documentare/studiare

“I backup sono da primitivi. I veri uomini scaricano i loro dati in un sito FTP in modo tale che gli altri possano dividerli” (Linus)

## Come è organizzato lo sviluppo open (1) aka la Cattedrale

- la Cattedrale (esr)
  - accentrato
  - organizzato (c'è il capo, il backup del capo...)
  - ci sono scadenze (prima dei concorrenti)
  - ci sono segreti, il software è chiuso...
  - c'è quello che hai e nessuno vuole darti una mano
  - ci sono concorrenti, non sempre che combattono solo a suon di idee

## Il Bazaar (4)

- C'è di tutto, si prende quel che serve
  - Un sacco di mattoncini sono già fatti, e se sono open si possono riutilizzare
  - “I bravi programmatori sanno cosa scrivere. I migliori sanno cosa riscrivere (e riusare)” (2#esr)
- Non ci sono “veri” nemici
  - Ci si concentra su come fare un buon software, non sul farlo prima di un concorrente o sugli uffici legali\*

\* c'è chi ci punta e... speriamo bene...

## Esempio di Bazaar

- <http://www.sf.net>
  - 96,000 progetti (tutti open)
  - forum, ML
  - CVS
- Ce ne sono altri
  - <http://www.freshmeat.net>
  - <http://savannah.gnu.org>
  - <http://www.wikipedia.org>

## Come sono organizzati i progetti open (1)

- Non c'è una vera e propria regola, esempi:
  - Linux: a piramide, nessuno sarebbe in grado di gestire da solo tutte le patch. “responsabili” che le provano/studiano/sistemano e man mano le passano più in altro
  - Samba: oligarchica, in pochi possono accedere al CVS in scrittura. Quando la “reputazione” di un esterno diventa suff. allora viene aggiunto
  - FreePOPs: i creatori hanno potere assoluto, gli esterni hanno accesso diretto solo sui “pezzi” fatti da loro

## Come sono organizzati i progetti open

- Forte e rigida struttura. Project Manager e via dicendo
- Gli utenti vedono il software quando è finito o quasi
- Il costo di debugging non viene abbassato dagli utenti prima della fatidica 1.0

## Come sono organizzati i progetti open (2)

- “Trattare gli utenti come co-sviluppatori è la strada migliore per ottenere rapidi miglioramenti del codice e debugging efficace” (6#esr)
- “Stabilisci una base di beta-tester e co-sviluppatori sufficientemente ampia, ogni problema verrà rapidamente definito e qualcuno troverà la soluzione adeguata” (8#esr)
- La parte più costosa dello sviluppo software è il debugging e la manutenzione, farsi aiutare dagli utenti non è una cattiva idea...

## Come sono distribuiti i software open

- Mezzi
  - La rete è il canale principale, una volta... per posta
  - Le riviste (non costa mettere sui CD software open)
- Modalità
  - Spesso (il feedback degli utenti serve a debuggare, e poi gli utenti vedono i problemi risolti in un tempo ragionevole)

## Come sono finanziati i progetti open

- Fino ad ora è stato facile dire che open è bello
- Storia del termine opensource:
  - era Free Software, che vuol dire sia *Libero* che *Gratis*
  - opensource venne creato per non essere ambiguo
- Quindi opensource non è gratis in tutti i sensi
  - Linux ha un “costo stimato” di 600M \$
  - Lo sviluppo dei sistemi che usate in lab iniziò “grossomodo” venti anni fa...

## Ma si muore di fame?

- Il software ~~open~~ si vende e qualche azienda ci fa un mucchio di soldi
- Il software è anche servizio, si può vendere quello
- Il modello di business proposto dal modello del software libero è meno redditizio ma (si spera) sostenibile
- Il modello di sviluppo software open produce software di qualità elevata (senza il si spera)

## Tirando le somme

- Nel modello di sviluppo open c'è:
  - una componente tecnica, il software open e' più facile da maneggiare, riusare, sistemare
  - una componente umana, l'utente è centrale, la collaborazione
  - una componente etica, la libertà, la cultura libera
  - una componente fortemente casuale ;)