



Open Source: i difetti diventano pregi

http://www.vbsimple.net/news/news_02.htm

Open Source non è solo la libera condivisione del codice sorgente (licenza GPL o simili), è soprattutto una filosofia, dai risultati entusiasmanti.

Open Source è un modo di condividere le conoscenze, tutti mettono le proprie capacità e conoscenze a disposizione degli altri; se il progetto è interessante, il numero molto alto delle persone che programmano e debuggano consente di sviluppare progetti notevoli e con crescita esplosiva.

Il progetto può essere spezzato e affidato a vari coordinatori, così ci si può occupare solo delle parti che ci interessano veramente.

Si consiglia vivamente la lettura di <http://www.agonet.it/mac/opensource/cattedrale.html> e degli Halloween I-II-III (il tutto è disponibile in Italiano).

È lo scritto in italiano di una persona, autore di un progetto (fetchmail per Linux), e spiega, con fatti e teorie, perché e come l'Open Source funziona, ed è, anzi, nettamente superiore al modo tradizionale di programmare.

Vediamo un semplice esempio?

Una persona può costruire 2 case, di bellezza 7, di qualità 7, di costo 8, in tot tempo. Venti persone possono costruire 100 case, di bellezza 8, di qualità 9, di costo 4, in tot tempo.

Questo perché una persona da sola dovrà fare il giardiniere, l'idraulico, l'arredatore, il muratore, ecc. e sarà efficace, e soprattutto, gli piacerà fare solo una delle cose.

In 20 persone ci sarà sicuramente qualcuno che gli piace fare ed è efficace per ogni ruolo.

In parole semplici Open source è anche *"ognuno faccia quello che gli piace e che è capace di fare, perché così si diverte e rende molto molto di più"*.

La filosofia Open Source fa' diventare degli enormi pregi quei difetti che erano tali, solo per l'incapacità di vedere oltre il proprio naso: il rifiuto della concorrenza, tipico dei progetti Closed Source.

Per iniziare, vi racconto di una mia esperienza involontaria e solitaria di Open Source, premetto che le mie basi di programmatore sono very Closed Source.

Ho iniziato scrivendo programmi in assembler per schede a microcontrollore usando un commodore C-64, per poi trasferirli al buio (senza poter cioè fare debug) sulla scheda dopo avere ricompilato il codice usando un assembler autoprodotta scritto con QBasic (quante a\$, b\$, aa\$ eheheh); più' Closed di così si muore.

Tempo fa ho iniziato un progetto (**FlexAgenda**) in Visual Basic, in perfetto stile Closed Source (con obiettivo il codice), scrivendo codice solo per me stesso, tenendone a mente tutta la struttura.

Risultato: qualità prodotto 4, qualità codice 5, robustezza 5, impegno richiesto 9, infatti passavo l'80% del tempo a gestire la struttura del codice (che rottura!!!).

Poi ho iniziato a non lavorarci per settimane, riprendere il lavoro per qualche giorno, poi di nuovo pausa per settimane e via di seguito per numerosi cicli.

Dopo ogni pausa mi ricordavo poco e niente del progetto, quindi sono stato costretto ad imparare a scrivere codice utilizzabile da un altro che avesse in mente poco e niente della struttura del codice (come me dopo le pause).

Insomma i cicli di pause-modifiche (cicli *open sourcizzanti*) mi hanno catapultato nell'Open Source, nello scrivere il codice per altri (non c'è differenza nello scrivere per altri rispetto allo scrivere per se stessi a distanza di tempo), avendo come obiettivo la struttura dei dati.

Mi è diventato normale chiedermi continuamente "*come posso scrivere il codice in maniera che sia semplice rileggerlo*"?

Risultato dei cicli *open*: qualità prodotto 7, qualità codice 7, robustezza 8 (un solo grosso bug in mesi di utilizzo e per giunta previsto, nonostante che la gestione degli errori fosse a dir poco grezza), impegno richiesto 7 (a tratti mi sono persino divertito!!).

Dopo questa esperienza sono giunto a queste conclusioni:

È falso che sia più rapido riscrivere che usare codice altrui, chiunque percorra un buon numero di cicli *open sourcizzanti* impara a scrivere codice (ri)leggibile.

Problema: è illeggibile?

Soluzione: riscrittura del programma con obiettivo i dati (gli altri) e non il codice (me stesso).

È falso che la leggibilità sia data dallo stile di scrittura.

È invece determinata dal porsi in ogni momento l'obiettivo e averne la necessità - cosa che non succede nei lavori Closed Source poiché i cicli *open sourcizzanti* sono troppo radi e quasi sempre ostacolati ed il progetto muore prima.

È falso che chi impara la programmazione in Visual Basic scrive codice peggiore di chi impara in C++, il programmatore VB, immerso nei cicli *open sourcizzanti*, produce codice di categoria nettamente superiore a un programmatore C++ Closed Source.

Qualsiasi progetto (o modifica) nasce in modalità Closed Source, ma può essere gestito con successo solo in modalità Open Source (in Closed Source il progetto diventa rapidamente ingestibile e palloso).

L'Open Source è basato su **Minima Massa Critica (MMC)**, **Basso Tempo di Risposta (BTP)**, **Anarchia Costruttiva (AC)**.

Quando queste cose funzionano insieme, il risultato è un'esplosione, di progetti, di codice, di alta quantità e qualità, che, una volta innescata continua a crescere fino al massimo consentito dallo spazio disponibile, dalla rapidità di comunicazione.

La sostituzione delle persone che escono dai progetti è rapidissima (come si dice, c'è la fila).

MINIMA MASSA CRITICA

Un progetto diventa vivo

È il primo pilastro dell'Open Source, ed è anche il nucleo di alcuni newsgroup, in primis

it.comp.lang.visual-basic.

Vediamo di spiegazzarci :)

Se avete frequentato quel NG, avreste notato che molti hanno fatto i complimenti trovandolo pieno di gente esperta, in gamba, disponibili, conoscitori di vari linguaggi; insomma un grande NG. Sapete perché e' così? Ci sono vari motivi.

C'è innanzitutto la spinta a primeggiare, lo spirito di gruppo, l'emulazione, e altre cosette simili. Questo fa sì che se c'è uno che risponde a decine di post, ecco che anch'io sono spinto a rispondere a più persone, se c'è uno che risponde con precisione e bravura, ecco che anch'io sono spinto a dare risposte migliori, se c'è uno che risponde anche agli inesperti, ecco che anch'io sono spinto a rispondere agli inesperti, ecc.

Quelli dell'Open Source di Linux lo chiamano *egocentrismo generoso* ma io preferisco definirlo aperto, cosciente, intelligente.

Ecco cos'è la *Minima Massa Critica*. Quando viene raggiunta succede che io spingo te, tu spingi un altro, un altro spinge un altro ancora e qualcun altro spinge me.

I risultati della reazione sono molto molto positivi per tutti ed una volta innescata, va' avanti da sola: anche se uno sparisce, viene rimpiazzato poiché nuove persone non vedono l'ora di partecipare.

Ma perché si inneschi questo è necessario un numero minimo di persone, che varia in base all'interesse che ha il progetto e alla velocità della comunicazione, altrimenti la reazione si spegne.

BASSO TEMPO DI PUBBLICAZIONE

La rapidità non è andare al limite

Come si ottiene un alta velocità di comunicazione?

È necessario lavorare almeno 23 ore al giorno?

Forse vi sto confondendo le idee, quindi parlerò ancora di un caso reale: del NG.

Molte persone hanno elogiato l'incredibile velocità di risposta e qualità, tale che, se dovesse fare concorrenza a un servizio di help di una ditta commerciale, il suddetto help verrebbe formattato dalla faccia della terra.

Come funziona la cosa?

Ci sono vari motivi.

Se io adesso ho cinque minuti liberi, mi sento spinto a rispondere ai vari post, perché so che la mia risposta arriverà in tempo utile (*Basso Tempo di Pubblicazione*).

Se il tempo di pubblicazione fosse alto chi me lo farebbe fare?

Tanto avrà già risposto qualcun altro. Tanto si sarà già arrangiato da solo?

Beh se si ha un alto numero di persone, per un semplice motivo statistico, nell'arco di 24h ci sarà sempre qualcuno che ha cinque minuti liberi e che si sente stimolato a rispondere.

Con tante persone, e con un basso tempo di pubblicazione si ottiene: basso sforzo del singolo, grande qualità globale, basso tempo di risposta globale o, meglio ancora, *il*

massimo con il minimo sforzo.

ANARCHIA COSTRUTTIVA

Gestire è un male, soffoca e incasina

Un vantaggio notevole dell'anarchia è che gli impegni (coordinamento, distribuzione, pubblicazione, ecc.) vengono suddivisi tra più persone. Alta velocità e basso sforzo.

Esempio di Open Source:

C'è un sito che funziona da raccoglitore e da coordinatore e rende disponibili i mezzi e gli strumenti necessari per rendere facile il lavoro in comune (strumenti di conversione per VB 3-4-5-6-100.000, strumenti per evidenziare le modifiche e aggiornare automaticamente i progetti, ecc.).

Chiunque può caricare e scaricare liberamente qualunque codice sorgente, progetti Open Source, sotto licenza GPL o simili.

Chiunque può registrarsi come gestore di un progetto Open Source, e solo il gestore potrà, liberamente, modificare il codice e il progetto Open Source (chiunque potrà comunque scaricare liberamente il progetto, il codice).

*Io, Dbg, carico e registro il progetto **FlexAgenda** (ehehehe).*

Altri trovano interessante la cosa e decidono di partecipare; essendo in tanti, il progetto viene diviso in sotto problemi e i sottoprogetti vengono caricati nel sito e registrati da altre persone.

Borrelli nota che da un po' di tempo Dbg non segue il progetto, lo gestisce male, fa' casino, e visto che Dbg fa finta di niente, decide di gestire lui il progetto.

Scarica il progetto **FlexAgenda**, lo modifica, lo carica nel sito Open Source come **Borrelli_FlexAgenda** e si registra come gestore.

Tutti i partecipanti insoddisfatti del progetto **FlexAgenda** passano al nuovo progetto.

*Fibia guarda tutti due i progetti, nota che sono complicati, astrusi, indigeribili, e decide di gestire un nuovo progetto, **FlexAgenda** per tutti (ehehehe).*

Scarica il progetto **FlexAgenda**, lo modifica, lo carica nel sito Open Source come **Simple_FlexAgenda** e si registra come gestore.

In maniera facile, indolore, rapida, si creano progetti in grado di interessare i vari tipi di interessi, attitudini, capacità, delle persone.

Alla fine il progetto migliore e forse vincente, sarà quello che avrà suscitato più interesse e non quello che un *dittatore* avrebbe deciso come valido.

I progetti **Borrelli_FlexAgenda**, **Simple_FlexAgenda** stanno crescendo, e il progetto **FlexAgenda** è un po' fermo a causa della notoria incostanza del suo gestore, ma ecco che in due mesi infuocati, prendendo spunto dai due progetti, il progetto viene rianimato dal suo gestore e dai partecipanti, accresciuto, recupera e sorpassa i progetti.

Intelligentemente i gestori e i partecipanti degli altri due progetti guardano al progetto **FlexAgenda** e recuperano il terreno perso.

Ognuno fa quello che gli piace fare, dove gli piace fare, con chi gli piace fare, quando gli

piace farlo e il risultato di questo sono sempre ottimi risultati.

E questi ottimi risultati sono disponibili a tutti, niente va' mai perso, tutto diventa almeno una base di partenza per qualcun altro.

Questo scritto è composto da mie e altrui esperienze e idee.

Chiunque voglia puntualizzare, criticare, suggerire, sostenere, o comunicare mi contatti su bepi@adria.it o bepii@libero.it.

[Giuseppe Della Bianca](#)

11 Dicembre 2000



[Torna all'indice degli Articoli](#)
