

[Home Page](#) [Informazioni](#) [Aiuto](#) 

Open Source Software: Cos'è? Perché usarlo?

http://www.vbsimple.net/news/news_01.htm

Cos'è il software "Open Source"?

"Open Source" è un nuovo termine, definito recentemente per descrivere quel software che è disponibile al pubblico in forma di codice sorgente e che non ha restrizioni di licenza che ne limita l'uso, la modifica e la redistribuzione. Il software Open Source è in genere coperto da copyright, e la sua licenza può contenere restrizioni intese a preservare il suo stato di Open Source, notizie sull'autore, oppure controlli di sviluppo. Il termine Open Source è stato registrato come marchio di certificazione dalla [Software in the Public Interest](#), la quale ha anche sviluppato la [definizione ufficiale di Open Source](#).

Il software Open Source è tipicamente sviluppato da un gruppo di programmatori sparpagliati in tutto il mondo, ma è anche sviluppato da università, agenzie di governo, imprenditori, consorzi e corporazioni *for profit*. Il software Open Source storicamente è stato molto intrecciato con gli ambienti UNIX e Internet, dove molte differenti architetture hardware devono essere supportate e le distribuzioni in forma sorgente sono l'unica via praticabile per completare questo porting multiplatforma. Nelle piattaforme DOS, Windows e Macintosh solo una piccolissima percentuale di utenti hanno a disposizione i compilatori, per cui il software Open Source è molto meno comune. Per una più dettagliata discussione sui modelli di sviluppo Open source leggi [The Cathedral and the Bazaar](#) di Eric Raymond.

Perché il software Open Source è rilevante nelle posizioni di mercato?

Il software commerciale tradizionale ha dietro enormi budget pubblicitari e di marchio, che gli danno un profilo molto alto, lontano dalla sua vera importanza. Il software commerciale tradizionale infatti è solo una piccola minoranza di tutti i software scritti; la parte veramente rilevante è sviluppata da e per conto di ogni sorta di organizzazione, per soddisfare i loro bisogni interni. Una parte significativa di tutti questi software sviluppati internamente, specialmente quelli commissionati dal governo USA, è sotto una licenza Open Source. È naturale conseguenza che la vasta maggioranza dei professionisti software non scrive software commerciale.

È difficile quantificare l'impatto medio o l'effettiva utilizzazione del software Open Source, ma è chiaro che entrambi sono molto ampi e, ancor di più, sottostimati dal pubblico generale. Oggi, molti prodotti Open Source stanno diventando sviluppati e supportati commercialmente. Allo stesso tempo il successo dei modelli di affari Open Source, l'alta produttività e la qualità dell'Open Source ha convinto la Netscape a sviluppare il suo browser sotto una licenza Open Source, almeno fino al 1998.

C'è un grande traffico dietro il movimento Open Source oggi; sarà interessante vedere quanto gli sviluppatori tradizionali di software commerciale dovranno evolversi per tenere

testa all'Open Source. Le industrie di computer hanno già visto una sorprendente crescita ed innovazione, come il primo microcomputer negli anni 70, e le produzioni di massa di PC nella metà degli anni 80, fino al presente, e la crescita di Internet nello stesso periodo. Il software che si basa sui modelli di sviluppo Open Source potrebbe essere la prossima nuova direzione.

Come differisce il software Open Source dagli altri software liberi?

Storicamente, molte persone che hanno sviluppato questo genere di software (come la Free Software Foundation) l'ha denominato "free software" - software libero - e la stampa l'ha chiamato anche freeware.

Purtroppo, entrambi questi termini hanno dimostrato di essere ambigui e vaghi. A molte persone il termine "libero" si riferisce al solo costo d'acquisto. Ma la cosa più importante del costo iniziale, tuttavia, è la libertà fornita dalla mancanza di restrizioni di licenza. Ci sono numerosi tipi di software che possono potenzialmente non avere costo, ma che non sono assolutamente Open Source a causa delle loro restrizioni o della mancanza di codice sorgente pubblico.

- **"ShareWare"**: spesso sviluppati da una singola persona, in genere non supportati e senza codice sorgente pubblicato, con la speranza di ricevere donazioni monetarie dagli utenti.
 - **Demo (Crippleware)**: nessuno o basso costo; sottospecie di un prodotto software commerciale, senza codice sorgente a disposizione liberamente, senza supporto, e, in genere con funzionalità mancanti.
 - **Software impacchettati**: forniti o aggiunti ad un software commerciale senza costo aggiuntivo, ma in genere sotto una licenza molto restrittiva and senza codice sorgente disponibile liberamente; un esempio è Microsoft Internet Explorer.
 - **"Consortium-ware"**: codice sorgente condiviso da un gruppo di aziende, ma non disponibile liberamente al pubblico; un esempio è il sistema Motif.
 - **Ad uso non commerciale**: software tipicamente scaricabile gratuitamente da Internet, forse anche con codice sorgente, ma al quale è stata posta una forte restrizione di licenza che ne limita l'uso; un esempio è il browser Netscape fino al 1998.
-

Alcuni tipi di licenze Open Source

- **BSD (Berkeley Source Distribution) License**: richiede notizia di copyright sull'autore.
- **GPL (General Public License)**: detta anche Copyleft, proibisce le restrizioni sui prodotti derivati e redistribuiti.
- **LGPL (Library General Public License)**: differisce dalla GPL poiché le librerie sotto tale licenza possono liberamente essere collegate a software proprietario.
- **Artistic License**: permette all'autore di mantenere il controllo di ulteriori sviluppi.
- **NPL (Netscape Public License)**: una nuova licenza nello spirito della GPL, ma mantiene il controllo e la proprietà del codice sorgente.

- **Dominio pubblico / Senza copyright / Senza restrizioni:** spesso utilizzata per algoritmi fondamentali allo scopo di incoraggiare un uso ampio, da parte del governo USA.

Alcuni esempi di software Open Source

Ci sono centinaia di prodotti maturi Open Source in ampio utilizzo oggi. Eccone alcuni dei più famosi:

- **BSD UNIX:** sviluppato dall'Università Californiana di Berkeley sotto la licenza BSD. Alcune versioni commerciali di UNIX sono direttamente basate sul codice sorgente di BSD oppure cambiano molto dal suo design iniziale.
- **Linux:** cresciuto in sei anni da un laureando finlandese, il progetto di ricerca ad uno dei più popolari sistemi operativi non-Microsoft. Originariamente solo per PC, adesso supporta Palm Pilots, i grossi supercomputer a 64 bit della Digital Alpha, e quasi tutti gli altri sistemi. Più efficiente, affidabile e moderno dei più tradizionali sistemi operativi. Fornito di compilatori, librerie e strumenti della [Free Software Foundation](#) e con una miriade di altri prodotti Open Source, la piattaforma Linux è la base di un piccolo ma di rapida crescita per industrie commerciali ed è un principio per l'innovazione software. Sviluppato da varie aziende commerciali e da migliaia di programmatori indipendenti, sparsi per tutto il mondo, sotto una licenza GPL.
- **Il linguaggio Perl:** il nastro di connessione di Internet; la base dei contenuti dinamici della maggior parte dei server Web, ampiamente utilizzato per lo scripting e l'automazione. Sviluppato dal Perl Institute sotto una Artistic License.
- **GCC:** Il *GNU C Compiler* è il preferito compilatore C e C++ nelle piattaforme UNIX, con porting a quasi tutti i sistemi operativi. Anche famoso come compilatore indipendente per il suo supporto back-end per quasi ogni processore esistente. Mantenuto dalla [Free Software Foundation](#), sotto GPL con il supporto di molti importatori software.
- **Il sistema X Window:** il più utilizzato sistema a finestre non Microsoft con avanzate caratteristiche di esecuzione remota indipendente dalla piattaforma, performance eccellenti ed un elevato grado di configurabilità e flessibilità. Sviluppato dal MIT e dall'X Consortium sotto una licenza simile a quella BSD, adesso controllato da The Open Group. Supporto per hardware PC ed altri miglioramenti sono attivamente controllati da [The XFree86 Project Inc.](#)
- **Internet Backbone:** l'Internet Engineering Task Force (IETF) preferisce le implementazioni di nuovi protocolli in Open Source per enfatizzare gli standard che sono già stati dimostrati all'opera in maniera neutrale aziendalmente (questo è stato il successo di Internet). I software con missione più critica su Internet sono sviluppati utilizzando il modello Open Source:
 - **BIND:** Il Berkeley Internet Name Daemon, comunemente chiamato *named* è utilizzato da quasi tutti i server di nomi di Internet, che agisce come un immenso database distribuito in tutto il mondo. Mantenuto dall'ISC (Internet Software Consortium).
 - **BSD Sendmail:** lo standard migliore degli MTA (Mail Transfer Agents - Agenti di trasferimento posta). Nessun competitore commerciale può avvicinarsi alle sue caratteristiche di base, alla comprovata sicurezza ed alla

sua robustezza. Gestisce il 75% di tutto il traffico di posta Internet, inclusa quella dei grandi siti come AOL, che da solo gestisce molti milioni di messaggi ogni giorno. Mantenuto dalla [Sendmail Inc.](#) sotto licenza BSD.

- o **INN (InterNet News Server)**: gestisce la maggior parte dei gruppi di discussione Usenet su Internet e molte Intranet aziendali. Mantenuto dall'ISC sotto licenza BSD.
- o **Apache**: potenza metà di tutti i web server (molto di più di Microsoft e Netscape assieme), per un totale di oltre un milione di web server fino al 1998. Sviluppato da **The Apache Group** con una licenza simile alla BSD.
- o **WU-FTPD**: il più popolare server FTP su Internet, utilizzato dalla maggioranza dei siti FTP. Sviluppato originariamente da Bryan O'Connor all'Università St. Louis di Washington, è adesso mantenuto dalla **Academ Consulting Services** e da molti volontari, con una licenza BSD.

Vantaggi di utilizzare software Open Source

- **Rischio ridotto.** L'aver il codice sorgente da ai clienti il controllo sugli strumenti dei loro affari. Quando uno sviluppatore di un prodotto Open Source innalza i suoi prezzi eccessivamente, aggiunge delle restrizioni alla licenza inaccettabili, e in qualche modo perde i suoi clienti. Questo fa sì che un'altra azienda che utilizza regolarmente codice Open Source possa iniziare un altro prodotto che risolve i problemi dello sviluppatore originale. I clienti possono anche scegliere di mantenere il software da loro stessi oppure di noleggiare qualcun altro che possa soddisfare i loro bisogni. Questo livello di controllo è spesso sconosciuto ai tradizionali modelli di software proprietario. Anche le aziende clienti con bassa leva finanziaria rischiano un tracollo a causa dei problemi interni ad un'azienda software tradizionale o dall'acquisizione di un'altra azienda.
- **Qualità.** Molti studi hanno dimostrato un critico vantaggio di serietà di software Open Source paragonato a molti altri prodotti commerciali. I modelli di sviluppo più effettivo, la decisiva possibilità di rivedere sia il codice sia il design ed il pregio di essere autore, contribuiscono a questo fenomeno. Alcune aziende vanno oltre, offrendo ricompense per la ricerca di errori nei loro software.
- **Trasparenza.** I software proprietari hanno molti angoli oscuri nei quali gli errori si possono nascondere. Il codice sorgente è cruciale per il debug e la comprensione di come un prodotto funziona. Nelle grandi compagnie, pochi impiegati hanno accesso al codice sorgente e gli impiegati che ne hanno accesso sono raramente disponibili ai clienti. L'accesso al codice sorgente è anche essenziale nella ricerca e correzione di buchi di sicurezza e comportamenti non previsti.
- **Unicità di caratteristiche.** Alcuni prodotti Open Source, inclusi molti di quelli visti precedentemente sono stati così decisivi poiché non esistono altri competitori commerciali.
- **Personalizzazione.** L'Open Source da ai clienti una maggiore possibilità di personalizzare il software per soddisfare i loro bisogni. Le grandi aziende possono risparmiare molti costi con una minima personalizzazione sviluppata su vasta scala. Le correzioni degli errori rilevati dai clienti diventano spesso parte dei pacchetti Open Source (diventando quindi mantenuti); questa possibilità non esiste con i

software commerciali tradizionali.

- **Licenza e costi favorevoli.** Per definizione i software Open Source hanno una licenza molto più flessibile di quella dei software proprietari. Questo può ridurre sostanzialmente i costi ed il lavoro richiesto per ulteriori installazioni, specialmente nelle aziende con costi legati al tempo o all'opera. Fornisce anche ai clienti maggiore flessibilità nel modo in cui il software è distribuito e può eliminare alcuni sforzi dei dettagli della licenza.

Quando non usare software Open Source

- **Caratteristiche errate.** Se un software Open Source non calza i requisiti di una'azienda, ed un software proprietario sì, è generalmente molto meno costoso utilizzare il prodotto proprietario che possiede già le caratteristiche giuste, piuttosto che effettuare molte grosse modifiche al prodotto Open Source. Vedi sopra la voce [Personalizzazione](#).
- **Necessità di supporto.** Alcuni prodotti Open Source mancano del tradizionale supporto a pagamento o telefonico. Tuttavia, in pratica, questo non è un problema molto serio; lo sarebbe se il prodotto fosse senza supporto e senza codice sorgente. Vedi la sezione [Supporto e software Open Source](#).
- **Piattaforma errata.** Molti software Open Source sono scritti originariamente per UNIX e Linux, a volte con Windows NT come bersaglio secondario. Altre piattaforme come i mainframe si presentano generalmente come una sfida troppo grande per effettuare un porting di un software Open Source. Il porting da UNIX a Windows NT può anche essere costoso, naturalmente dipendentemente dalla natura del prodotto. I vantaggi dell'Open Source, tuttavia, possono giustificare il cambio di piattaforma.
- **Mancanza di staff tecnico.** I prodotti Open Source richiedono molto esperienza nell'installazione, così come nella configurazione e compilazione del codice sorgente distribuito. I benefici dell'avere il codice sorgente sono molto ridotti in quelle aziende senza esperienza di sviluppo software. L'effettiva personalizzazione di software Open Source richiede una conoscenza manageriale. Tuttavia, oltre questi problemi tecnici, ogni azienda può godere dei benefici dei software Open Source.
- **Inerzia.** Se i sistemi esistenti proprietari sono ben supportati ed il funzionamento è adeguato non è conveniente cambiarli. Tuttavia è importante rivisitare alcune decisioni periodicamente, che le tecnologie frequentemente rendono obsolete.

Ragioni dubbie per evitare i software Open Source

Queste ragioni sono spesso utilizzate per evitare i prodotti Open Source, ma le quali non restano valide sotto un'attento scrutinio.

- **Paura dello sconosciuto.** Come visto sopra, negli [esempi di software Open Source](#), molte organizzazioni si spostano sui software Open Source se coloro che prendono le decisioni aziendali sono convinti o meno. Questo problema sta diventando facile da superare grazie al miglioramento del mercato ed alla pubblicità degli sviluppatori Open Source.

- **Abbiamo a disposizione grossi investimenti per software commerciali.** Questa è solo una forma di sperperar denaro. Le licenze software, così come l'affitto degli spazi per gli uffici, sono spese, non investimenti. Se un altro prodotto soddisfa maggiormente i bisogni aziendali allo stesso o minore costo futuro, le spese passate per il prodotto inferiore non legittimano questa decisione.
- **Abbiamo bisogno di incolpare qualcuno in caso di problemi.** Mentre questa può sembrare una ragione valida in teoria, l'utilizzo di un prodotto funzionante è molto più effettivo di cercare un fornitore di software negligente da querelare in caso di problemi futuri. Quasi tutte le maggiori compagnie di software impongono un'accettazione della licenza finale per l'utente che assolve loro da ogni concepibile colpa proveniente dai problemi nei loro software. Microsoft e IBM hanno enormi, superpagati staff legali che assicurano loro che le lamentele dei clienti non possono avere successo.

Supporto e software Open Source

Contratti di supporto commerciali sono in genere in cima alla lista delle ragioni per le quali le aziende utilizzano software Open Source. Il supporto a pagamento è una valida motivazione, ma spesso è disponibile anche nei prodotti Open Source ed è in genere in sovrastima.

- **Contratti di supporto sono disponibili per i maggiori prodotti Open Source.** Per esempio Red Hat e Caldera forniscono contratti di supporto telefonico per le loro distribuzioni Linux. Cygnus, Collective Technologies e molte piccole compagnie di consultazione fanno i loro affari supportando e migliorando i software Open Source. E un numero sempre maggiore di prodotti Open Source sta diventando commercialmente mantenuto pur mantenendo il loro stato di Open Source. Ne sono esempi Linux, Sendmail, BIND e DHCP.
- **Supporto E-Mail e notizie Usenet sono estremamente efficaci.** Quasi tutte le organizzazioni che sviluppano prodotti Open Source offrono supporto e-mail gratuito, mailing list e/o gruppi di discussione Usenet. Questi meccanismi sono molto più efficienti e meno costosi del supporto telefonico. La stessa comunità degli utilizzatori Linux si è aggiudicata il "Best Technical Support" - miglior supporto tecnico - nel 1997 da InfoWorld, basata su un'inchiesta che ha rivelato molta insoddisfazione nei supporti offerti dai tradizionali fornitori di software.
- **Il supporto per i software commerciali tradizionali è spesso limitato e povero in qualità.** Microsoft, per esempio, provvede solo circa 900 numeri per tutti i suoi prodotti. La maggior parte dello staff di assistenza sui prodotti comuni ha una conoscenza dei problemi di utilizzo per principianti ed ha difficoltà a rispondere a domande che vanno oltre a quello scritto nel manuale d'uso. Molte aziende di software hanno considerato elevati e superflui i costi di supporto tradizionale telefonico, al punto di toglierli. La qualità spesso aggrava i risultati.

Un ringraziamento particolare all'autore dell'articolo: [Niemi di tux.org](http://www.niemi.fi/tux.org)

Documento originale: [Open Source Software: the Customer Case](#)

Traduzione di [Fibia FBI](#)

11 Dicembre 2000

Rivisitato il 24 Marzo 2002



[Torna all'indice degli Articoli](#)
