

**CONVENZIONE INTERBANCARIA
PER I PROBLEMI DELL'AUTOMAZIONE
(CIPA)**

GRUPPO DI LAVORO

**LA VALUTAZIONE DELLE OPPORTUNITA' E DEI
COSTI CONNESSI CON L'ADOZIONE DEI PRODOTTI
OPEN SOURCE**

In relazione alle indicazioni contenute nel Piano delle attività della CIPA per il periodo 1.1.2002 – 30.6.2003, il gruppo di lavoro “Le valutazioni delle opportunità e dei costi connessi con l’adozione dei prodotti Open Source”, rassegna il proprio rapporto, che fa riferimento alla situazione in essere alla fine del 2002.

La Segreteria della Convenzione desidera ringraziare i componenti del gruppo di lavoro, di seguito indicati, per la collaborazione prestata e il contributo fornito nello svolgimento delle attività del gruppo:

Ebe	BULTRINI	Banca d'Italia (coordinatrice)
Maria	BIANCO	Banca d'Italia – Segreteria CIPA
Roberto	CIPOLLA	Banca d'Italia
Piero	PIPERNO	ABI
Leopoldo	PALOMBINI	Banca Nazionale del Lavoro
Cristina	MASSABO'	San Paolo IMI
Marco	BERNARDINI	Banca Monte dei Paschi di Siena
Giorgio	TOMEI	UniCredito Italiano
Roberto	DI CASTRO	Banca Antonveneta

IL SEGRETARIO
(A.M. Contessa)

INDICE

Sintesi	5
Premessa	9
1 Alcuni cenni sulla storia dell'open source software	10
1.1 Gli aspetti giuridici dell'Open Source.....	12
1.1.1 I riferimenti giuridici	12
1.1.2 Le licenze Open Source	13
1.2 L'organizzazione e il modello di sviluppo dei progetti Open Source.....	14
1.3 Le motivazioni dei partecipanti alle comunità Open Source.....	17
1.4 La cultura della condivisione delle informazioni.....	18
2 Classificazione dei prodotti open source e loro diffusione	20
2.1 Il software di base e il software infrastrutturale.....	21
2.2 Middleware.....	26
2.3 Le applicazioni di <i>office</i>	27
2.4 Il software applicativo <i>enterprise</i>	29
2.5 Linux <i>embedded</i>	30
2.6 Il software di " <i>frontiera</i> ".....	30
3 Le aree di possibile utilizzo con riferimento alle banche	32
3.1 Ambiente Mainframe.....	32
3.1.1 Mainframe Tradizionale	32
3.1.2 Mainframe Unix	32
3.1.3 Mainframe Linux	33
3.2 Ambienti distribuiti.....	34
3.2.1 Server infrastrutturali	34
3.2.2 Server applicativi	35
3.2.3 I server di back-end	36
3.2.4 Alcune considerazioni	37
3.3 Postazioni <i>Client</i>	37
4 Le ragioni a favore dell'open source	40
4.1 Bassi Costi.....	40
4.2 Aderenza agli standard.....	40
4.3 Affidabilità, qualità e adattabilità.....	41
4.4 Flessibilità.....	41
4.5 Indipendenza dai fornitori.....	42
4.6 Protezione degli investimenti.....	42
4.7 Supporto.....	42
5 Profili critici del software open source	43
5.1 Modello di <i>business</i>	43
5.2 Assenza di responsabilità legale sul software.....	44
5.3 Pianificazione e controllo dei progetti.....	44
5.4 Difficoltà nell'individuare l'offerta di prodotti.....	44
5.5 Disomogeneità dei prodotti software.....	44
5.6 Competenze tecniche.....	45
5.7 Ridotta disponibilità di applicazioni commerciali.....	45
6 La sicurezza	46
7 Analisi dei costi connessi ai prodotti open source	49
7.1 Identificazione dei requisiti.....	49
7.2 Il Total Cost of Ownership.....	50
8 Le raccomandazioni e le best practice	54
8.1 Le <i>best practice</i>	54
8.1.1 La raccolta delle informazioni	54
8.1.2 Le risorse umane	55

8.1.3	<i>Le risorse tecniche</i>	56
9	<i>Il futuro dell'open source software</i>	57
10	<i>Allegato A - L'orientamento delle Pubbliche Amministrazioni</i>	59
10.1	Indagine sull'Open Source Software e settore pubblico in Europa.....	59
10.2	La situazione attuale in Europa.....	60
10.2.1	<i>Francia</i>	60
10.2.2	<i>Germania</i>	62
10.2.3	<i>Spagna</i>	63
10.2.4	<i>Italia</i>	64
10.2.5	<i>Svezia</i>	65
10.2.6	<i>Belgio</i>	65
10.2.7	<i>Commissione europea</i>	66
10.3	La situazione in alcuni Paesi extraeuropei	66
10.3.1	<i>Gli Stati Uniti</i>	66
10.3.2	<i>L'America del Sud</i>	67
10.3.3	<i>L'Asia</i>	68
11	<i>Allegato B - Le strategie dei maggiori vendor</i>	69
11.1	IBM.....	69
11.2	Microsoft	70
11.3	Sun Microsystems	71
12	<i>Glossario</i>	73
	<i>Bibliografia</i>	75

Sintesi

Il documento del gruppo di lavoro CIPA “La valutazione delle opportunità e dei costi connessi con la diffusione dei prodotti Open Source” si compone di nove capitoli e due allegati. Dopo un inquadramento generale della materia attraverso cenni storici sull’evoluzione del software Open Source e una descrizione dell’organizzazione delle comunità Open Source e del modello di sviluppo dei relativi progetti, si procede a una classificazione dei prodotti e della loro attuale diffusione. Vengono poi passate in rassegna le aree di possibile utilizzo con riferimento al sistema bancario. Segue una dettagliata analisi dei punti di forza e di debolezza connessi con l’adozione del software della specie. Per la loro complessità, vengono trattate a parte le problematiche relative alla sicurezza. Si analizzano, quindi, i costi diretti e indiretti collegati all’utilizzo dei prodotti Open Source, si esprimono alcune raccomandazioni del gruppo di lavoro e le “best practices” da tener presenti qualora si intenda adottare prodotti della specie. Lo studio si conclude con l’indicazione di alcune iniziative di enti pubblici e privati e con le previsioni degli operatori del settore informatico che potranno influenzare il futuro dell’Open Source. Nell’allegato A) si riporta un’ampia ricognizione sulla diffusione dell’Open Source nelle PP.AA. europee ed extraeuropee, nell’allegato B) la sintesi delle prospettive di alcuni fornitori con riguardo al tema trattato.

1. L’Open Source, ovvero il software distribuito con il codice sorgente, costituisce indubbiamente uno dei fenomeni più interessanti nell’attuale panorama dell’Information and Communication Technology, coinvolgendo, accanto agli aspetti più propriamente tecnologici e organizzativi, anche profili economici, giuridici e sociali.

Le origini dell’Open Source risalgono agli anni ’60, con la comparsa dei primi computer commerciali, che venivano forniti con un corredo di software, comprensivo del codice sorgente, liberamente modificabile e scambiabile tra gli utenti. Successivamente, con lo sviluppo dell’Information Technology, il software si è venuto configurando come un bene immateriale avente autonoma rilevanza rispetto all’hardware; ciò ha condotto, a partire dagli anni ’70, all’attuale modalità di distribuzione del software “proprietario” protetto da una licenza d’uso che non consente interventi sul codice sorgente.

L’idea politica sottostante la coniazione del termine “free software”, inteso in contrapposizione al software proprietario, venne introdotta da Richard Stallman che, nel 1984, scrisse il Manifesto GNU (GNU Not Unix), nel quale vengono affermati i diritti dell’utente, e predispose la GNU Public License (GPL) ovvero lo strumento che garantisce la tutela legale di tali diritti.

Negli anni ’90, la diffusione di PC e di collegamenti a Internet a costi accessibili e l’invenzione del *World Wide Web* si riflessero in un allargamento della base di programmatori Open Source. In tale contesto si colloca l’attività di Linus Torvalds, ideatore del sistema operativo Linux. Torvalds rese pubblici i sorgenti via Internet suscitando l’interesse di molti programmatori che collaborarono per migliorarlo ed espanderlo. La diffusione di Internet ha determinato negli anni successivi un crescente consenso nei confronti del fenomeno Open Source, che si propone, in molteplici ambiti, come un’alternativa ai sistemi proprietari.

Il processo che regola la nascita e l’evoluzione di un progetto Open Source è reso particolarmente complesso dalla dinamicità con la quale mutano le categorie di partecipanti ai gruppi di progetto, la quantità, la tipologia delle risorse tecniche disponibili e le fonti di finanziamento. Un progetto nasce, di norma, dall’idea di un singolo individuo e per ragioni che vanno da un interesse di carattere scientifico, alla necessità di migliorare il proprio lavoro o di

creare un prodotto funzionalmente equivalente a uno proprietario. Il progetto si sviluppa poi all'interno di un gruppo di persone che condividono un interesse sulle medesime tematiche. La diffusione dei risultati dello sviluppo iniziale di un progetto Open Source (codice sorgente del software, documentazione, finalità del progetto, standard di riferimento etc.) tramite Internet si è dimostrato un fattore di stimolo alla nascita di comunità di sviluppo Open Source. Le motivazioni che sostengono il lavoro dei partecipanti alle comunità di sviluppo rendono queste assimilabili a quelle scientifiche, con le quali condividiamo, oltre la metodologia di lavoro, le classiche forme di riconoscimento sociale: l'eponimia, la citazione, i premi.

Di carattere economico sono invece le motivazioni dei distributori dell'Open Source e dei fornitori che hanno integrato i prodotti della specie nella propria offerta commerciale.

L'ingresso nel mondo Open Source di attori animati da interessi commerciali si è rivelato utile ai fini della diffusione del fenomeno, favorendo lo sviluppo di componenti ritenute poco interessanti dalle comunità.

2. Ai fini della classificazione dei prodotti Open Source il gruppo di lavoro si è avvalso di una molteplicità di fonti: oltre a consultare siti *web* che ospitano i progetti e contengono cataloghi aggiornati dei prodotti, sono stati analizzati i dati e le informazioni pubblicati da fornitori, analisti e stampa specializzata.

I progetti che presentano profili di maggior interesse per il settore bancario - in quanto maggiormente consolidati, con un numero consistente di sviluppatori, di utenti attivi e di partecipanti alle *mailing list* e ai *newsgroup*, sono quelli che riguardano, nell'ordine: il software di base e infrastrutturale, il *middleware*, le applicazioni di *office* e il software applicativo.

Per ciascuna delle famiglie di software indicate sono state fornite descrizioni dei prodotti che si sono già affermati o che hanno trovato un evidente interesse presso gli utenti (es. Linux, Apache, MySQL).

3. Le aree di possibile utilizzo dei prodotti Open Source nel sistema bancario sono state individuate avendo presente che i sistemi informativi delle banche sono costituiti da ambienti tecnologici e applicativi estremamente diversificati.

Con riferimento ai diversi ambienti (mainframe – elaboratori intermedi – postazioni client) si sono indagate le possibilità di utilizzo di prodotti Open Source.

Per quanto attiene al Mainframe Linux si rilevano le opportunità offerte, in termini di possibili riduzioni dei costi gestionali, dal consolidamento in unico ambiente di un elevato numero di server e dalla possibilità di realizzare applicazioni *mission critical* architetturealmente innovative sullo stesso hardware che ospita i dati *enterprise*.

Per quanto concerne gli elaboratori intermedi, sino a ora ambiente di elezione dell'Open Source, i possibili ambiti di utilizzo sono molteplici; l'offerta comprende prodotti leader di mercato; tutto ciò, unito alla disponibilità di numerosi prodotti commerciali su piattaforma Linux, di fatto amplia notevolmente la capacità di scelta dell'utente e crea positive forme di concorrenza.

Con riferimento, infine, alle postazioni client, le novità sono numerose: il mercato di Linux registra una crescita, così come confermato anche da alcuni dei principali fornitori, e i prodotti Open Source hanno raggiunto un buon livello di completezza funzionale. Sebbene per tale comparto sia prevedibile una certa resistenza degli utenti ad accettare cambiamenti nelle modalità operative, vengono descritti alcuni scenari in cui l'adozione dei prodotti Open Source può offrire vantaggi economici e benefici in termini di maggiore indipendenza da un unico fornitore.

4. L'adozione in ambito bancario di prodotti Open Source può consentire il conseguimento di diversi vantaggi, purchè si accompagni a coerenti interventi sull'organizzazione interna dell'azienda e si inserisca in una adeguata strategia I.T.

Uno dei primi vantaggi connessi con l'adozione di prodotti Open Source è indubbiamente costituito dai *bassi costi* di acquisizione. I prodotti Open Source sono, infatti, distribuiti gratuitamente o a costi "una tantum" molto contenuti. Va inoltre sottolineata l' *aderenza agli standard* di molti di tali prodotti, fondamentale per tutti quei soggetti (sistema bancario, Pubbliche amministrazioni) che devono interagire con una platea molto vasta di interlocutori esterni. Un ulteriore vantaggio è spesso costituito dall' *affidabilità, qualità e adattabilità* dei prodotti. Il modello di sviluppo Open Source rende infatti possibile a un gran numero di utenti la verifica del corretto funzionamento del software rispetto a un campione molto vasto di condizioni diverse fra loro e non prevedibili a priori, cosicché il prodotto finale può risultare più stabile e affidabile anche di taluni prodotti proprietari. La disponibilità del codice sorgente e la diffusa conoscenza delle tecnologie utilizzate nello sviluppo di prodotti Open Source, quali i linguaggi di programmazione e le tecniche di comunicazione, garantiscono altresì ampia *flessibilità* nell'uso del software, *indipendenza da singoli fornitori* e accrescono il livello di competizione nel mercato del software.. Rispetto ai prodotti proprietari, le cui modifiche richiedono spesso l'aggiornamento di altri componenti software e in alcuni casi il rinnovo delle piattaforme hardware, l'utilizzo di prodotti Open Source consente di programmare con maggiore autonomia l'utilizzo di versioni successive e, al limite, di non utilizzarle affatto con il vantaggio di *salvaguardare gli investimenti* realizzati.

5. Sul fronte dei rischi e dei possibili aspetti problematici, emerge in primo luogo la mancanza di una consolidata teoria economica sui modelli di *business* centrati sul software Open Source: ciò impedisce una adeguata valutazione della solidità delle nuove aziende la cui attività principale si concentra sui prodotti della specie, e, nel caso dei grandi fornitori, rende difficoltoso individuare il reale valore strategico della loro politica commerciale a favore di tali prodotti.

Da un punto di vista più strettamente giuridico si pone il problema, stante l'assenza di un soggetto titolare della proprietà intellettuale del software, di individuare le *responsabilità in caso di danni* diretti o indiretti causati dal prodotto.

Non vanno poi trascurate le difficoltà che possono emergere nell'attività di *pianificazione* e di *controllo dei progetti*, laddove gli stessi non abbiano raggiunto elevati livelli di stabilità .

Difficile e complessa può inoltre rivelarsi *l'individuazione dell'offerta*, atteso che mentre le informazioni sull'offerta dei prodotti software proprietari raggiungono gli utenti attraverso i molteplici canali attivati dalle aziende fornitrici nella loro attività di marketing, quelle sui prodotti Open Source, pur essendo spesso numerose, sono in genere pubblicate esclusivamente su siti *web* specialistici e, pertanto, sono meno agevolmente reperibili.

Sotto un altro profilo, il vantaggio di poter apportare modifiche a un programma con grande facilità e senza vincoli legali determina il rischio della *proliferazione di versioni disomogenee*, e in alcuni casi incompatibili, dello stesso prodotto Open Source.

Come sopra accennato, non vanno trascurati i costi connessi con la necessità, in taluni casi, di riorientare le *competenze tecniche* presenti in azienda.

Infine va segnalata la *ridotta disponibilità dell'offerta* di applicazioni commerciali in ambiente Open Source (Linux) rispetto a quella disponibile su piattaforme proprietarie, in particolare per gli ambienti MS Windows.

6. Il dibattito sulla sicurezza dei prodotti Open Source è particolarmente effervescente e vede contrapposte due "scuole di pensiero". La prima, richiamandosi al metodo scientifico di verificabilità dei processi, tende a ritenere il software Open Source più sicuro, la seconda, di contro, vede nella disponibilità dei codici sorgenti del software una possibile minaccia all'integrità dei sistemi informatici. Il gruppo di lavoro, raccogliendo molteplici dati fondati su evidenze empiriche - verificati sui siti che mantengono database aggiornati sulle vulnerabilità, i virus e gli incidenti occorsi negli ultimi anni - ha cercato di fornire un quadro il più possibile ampio sulle problematiche di sicurezza poste dal software, sia proprietario sia Open Source.

7. Relativamente al profilo della valutazione dei costi dei prodotti Open Source, il gruppo di lavoro si è inizialmente soffermato sui requisiti qualitativi da considerare nell'effettuazione di ponderate scelte. Per ciascuno di tali requisiti sono stati forniti alcuni possibili criteri di valutazione con riguardo ai prodotti Open Source. Al fine di consentire il confronto, in relazione a una specifica esigenza, di soluzioni basate su software Open Source rispetto a soluzioni proprietarie, è stato elaborato un modello per la valutazione dei costi diretti e indiretti (Total Cost of Ownership) di un determinato sistema informatico. In particolare, con riferimento alle principali fasi di ciclo di vita del prodotto (raccolta e valutazione delle informazioni, sviluppo della soluzione, messa in esercizio, training, utilizzo, manutenzione e indisponibilità del sistema) sono state fornite alcune indicazioni sull'incidenza dei costi in due distinti scenari. Il primo mette a confronto le due opzioni possibili per l'acquisizione di un pacchetto software (acquisto di un prodotto commerciale, adozione di un prodotto Open Source); il secondo scenario confronta i costi di uno sviluppo proprietario rispetto a quelli di uno sviluppo condotto all'interno di una comunità Open Source. Le indicazioni sono basate principalmente su considerazioni di carattere generale relative alle peculiarità del software Open Source; una puntuale valorizzazione dei costi deve essere, invece, condotta all'interno di ciascuna azienda in quanto i costi diretti e indiretti del software sono fortemente influenzati dall'organizzazione interna e dalle specificità del mercato.

8. Ritenendo l'Open Source una realtà ormai presente nel mercato del software e in grado di offrire in alcuni settori vantaggi economici e funzionali, il gruppo di lavoro ritiene che l'adozione di prodotti Open Source presuppone che sia stato individuato l'ambito di adozione e che siano valutati i costi e i benefici attesi nei confronti del software proprietario; occorre in ogni caso prestare attenzione alla portata innovativa sul piano culturale, organizzativo e tecnico del fenomeno. Infine vengono richiamate alcune *best practices* che possono ridurre i costi diretti e indiretti connessi con l'adozione delle tecnologie Open Source. In particolare, ci si sofferma sulla necessità di creare un inventario dei prodotti della specie, delle informazioni rilevanti e di eventuali test e *benchmark* condotti. Si affrontano, infine, i temi relativi all'impegno delle risorse umane e ai necessari presidi tecnici.

9. In conclusione del lavoro si riportano alcune iniziative di enti pubblici e privati nonché gli orientamenti degli operatori del settore che possono influenzare il futuro dell'Open Source. Tra gli elementi positivi particolare rilevanza ricoprono: il forte interesse mostrato dalle Pubbliche Amministrazioni, le cui iniziative nel settore potrebbero creare la massa critica necessaria a garantire all'Open Source una durevole presenza nel mercato; le previsioni di crescita di Linux che, seppure con forti scostamenti nelle stime, sono ritenute in aumento dagli analisti. Tra gli elementi che possono influire negativamente si cita il diffuso ricorso alla brevettazione del software, il rischio di chiusura dei prodotti più noti in versioni commerciali e, infine, le iniziative correlate allo sviluppo di meccanismi di protezione forte che potrebbero limitare l'uso di prodotti non "certificati" su alcuni hardware.

Premessa

L'Open Source Software è divenuto di recente uno dei fenomeni più interessanti nel panorama dell'Information and Communication Technology. La discussione avviatasi sul tema è molto accesa e ricorda quella che ha accompagnato la prima fase della diffusione di Internet.

Per comprendere e analizzare le tematiche intorno alle quali si è sviluppato il dibattito è necessario premettere alcune considerazioni di carattere generale sulle natura del software e sui conseguenti riflessi sulla sua produzione e commercializzazione.

Se assumiamo con C. Shapiro e H.R. Varian ⁽¹⁾ che "tutto ciò che può essere digitalizzato, ovvero rappresentato come sequenza di bit, è informazione", il software è informazione pura la cui economia è caratterizzata da elevati costi di produzione e insignificanti costi di riproduzione e di distribuzione. La domanda del software è influenzata dal livello di diffusione dello stesso, pertanto il suo valore cresce all'aumentare del numero di persone che ne fanno uso.

Il software, in qualità di prodotto indispensabile all'uso dell'hardware, presenta un enorme potenziale commerciale. Nel 2001 il solo software "pacchettizzato" ha dato vita a un'industria del valore di circa 196 miliardi di dollari (fonte OECD 2002) contribuendo positivamente alla crescita economica dei paesi nei quali è presente e divenendo uno degli elementi che maggiormente concorre ad aumentare il distacco tecnologico (il c.d. *digital divide*) dei paesi che non possono sopportarne gli alti costi di approvvigionamento, con conseguenze negative sul complessivo sviluppo economico e sociale di tali paesi.

Si evidenzia, quindi, come il software - per la sua duplice natura di bene immateriale e di strumento tecnico per il funzionamento della gran parte degli apparati elettronici - si inserisca in un dibattito generale che cerca nuove modalità di coesistenza tra il diritto alla proprietà intellettuale, l'innovazione, la competizione e lo sviluppo sociale ed economico.

E' in questo scenario che si colloca il dibattito avviatosi sull'Open Source che, grazie a un processo di contribuzione volontaria e gratuita allo sviluppo del software, resa possibile dalle favorevoli condizioni create dalla capillare diffusione di Internet, ha di fatto aperto un fronte di contrapposizione al modello di *business* proprietario dell'industria del software.

La definizione Open Source ⁽²⁾ sta a indicare software che viene distribuito con il codice sorgente protetto da licenze d'uso che ne consentono la copia, la modifica e la libera distribuzione; è prelevabile da Internet a titolo gratuito ed è il risultato del lavoro collaborativo di migliaia di programmatori geograficamente dispersi e, di solito, volontari che operano all'interno di comunità virtuali interconnesse via Internet.

Le novità introdotte dall'Open Source sono, quindi, molteplici e spaziano dagli aspetti più propriamente tecnologici e organizzativi a quelli ideologici, sociali e giuridici coinvolgendo importanti aspetti economici legati all'industria del software.

¹ C. Shapiro e H.R. Varian "Information Rules. A strategic Guide to the Network Economy", Harvard Business School Press, 1999.

² Il termine *Open Source Software* ha sostituito, nel 1998, l'espressione *free software* che, per l'ambiguità del significato del termine *free*, è stata a volte mal interpretata come sinonimo di software gratuito. In realtà il termine *free software*, inteso come software sul quale l'utente ha la libertà di operare, non ha niente a che vedere con il prezzo ma con la libertà connessa con l'utilizzo del software stesso. Lo studio non prenderà in esame il software c.d. *freeware* in quanto si riferisce in genere a quei pacchetti software per i quali è consentita la distribuzione ma non la modifica non essendo disponibile il relativo codice sorgente, né il software c.d. *shareware* che segue le medesime regole di distribuzione del *freeware* ma a differenza di quest'ultimo richiede il pagamento di una licenza d'uso per il suo utilizzo.

1 Alcuni cenni sulla storia dell'open source software

Quando, all'inizio degli anni '60, i primi grandi computer commerciali fecero la loro comparsa nei centri di calcolo, l'IBM e gli altri produttori fornivano tali sistemi con un corredo di software, comprensivo del codice sorgente, che poteva essere liberamente modificato e scambiato tra gli utenti. Con la trasformazione del software da bene immateriale, composto da complessi calcoli e algoritmi matematici, a prodotto commerciale da cui trarre un profitto separato da quello dell'hardware, si impose a metà degli anni '70 la modalità di distribuzione del software "proprietario", cioè di software protetto da una licenza che non concede agli utenti la possibilità di intervenire sul codice sorgente per poterlo modificare e ridistribuire.

Gli anni '60 coincidono anche con la nascita di importanti centri di ricerca sulla scienza dell'informazione sia pubblici che privati ⁽³⁾. La diffusione presso tali strutture dei minicomputer della serie PDP della Digital Equipment Corporation (DEC), favorita dalla loro relativa economicità, flessibilità e potenza, fu determinante per lo sviluppo dell'informatica. La stessa ARPAnet ⁽⁴⁾ fu costituita, per la maggior parte della sua durata, da una rete di macchine DEC e favorì lo scambio di informazioni tra i ricercatori, ovunque essi si trovassero, con velocità e flessibilità senza precedenti, dando un forte impulso allo sviluppo del lavoro di collaborazione e accelerando enormemente il ritmo e l'intensità del progresso tecnologico.

Negli stessi anni, a cura di alcuni ricercatori dei Bell Labs di AT&T (Thompson e Ritchie) nasceva una linea di sviluppi finalizzati alla realizzazione di un nuovo sistema operativo che avesse le caratteristiche di portabilità su piattaforme hardware diverse. Dobbiamo al lavoro di Thompson e Ritchie l'invenzione del linguaggio C, la definizione dei principi fondamentali dei sistemi operativi e lo sviluppo del sistema operativo Unix.

Nel 1974 Thompson e Ritchie pubblicarono sulla prestigiosa rivista dell'Association for Computing Machinery (ACM) lo "Unix Programmer's Manual". Tale articolo suscitò un enorme richiesta del codice del sistema Unix da parte delle Università e dei centri di ricerca. Alla fine degli anni '70 Unix era di fatto utilizzato e conosciuto negli ambienti accademici di tutto il mondo.

L'AT&T, dovendo evitare qualsiasi conflitto con il decreto emesso dalla commissione Antitrust ⁽⁵⁾, decise di licenziare il software senza fornire alcuna forma di supporto al software o agli utenti.

La crescente comunità Unix senza tali garanzie fu forzata a operare autonomamente; questo segnò la nascita di un movimento i cui principi erano quelli di "cooperare scambiando idee, informazioni, programmi, correzioni ai malfunzionamenti". Furono creati diversi *user group* cui partecipavano, tra gli altri, numerosi enti universitari di varie nazioni.

Il rifiuto di riconoscere *royalties* all'AT&T per l'utilizzo di Unix si risolse in una profonda divergenza negli sviluppi di tale sistema ⁽⁶⁾.

³ Tra questi i più innovativi furono l'Artificial Intelligence Lab del MIT, il laboratorio di Intelligenza Artificiale dell'Università di Stanford (SAIL), l'Università Carnegie-Mellon (CMU), la General Electric, la Xerox e i Bell Labs di AT&T.

⁴ ARPAnet (1969) è stata la prima rete transcontinentale di computer ad alta velocità. Ideata e realizzata dal Ministero della Difesa statunitense (DoD) come esperimento nelle comunicazioni digitali, crebbe fino a diventare un collegamento tra centinaia di università, enti legati al DoD e laboratori di ricerca.

⁵ L'AT&T nel 1949 era stata accusata dalla divisione Antitrust del Dipartimento di Giustizia di aver violato lo "Sherman Antitrust Act". A conclusione della causa era stato fatto divieto all'AT&T di operare in altri settori che non fossero quelli dei servizi di telefonia.

⁶ Di fatto si aprì un'altra linea di sviluppo dello Unix presso il Computer Science Research Group (CSRG) dell'Università della California Berkeley che già da tempo operava per il miglioramento del sistema operativo Unix e

Negli anni successivi molte versioni dello Unix erano presenti sul mercato, alcune derivate dalle versioni AT&T altre dalla linea di sviluppo curata dall'Università di Berkeley (BSD Unix), ormai tra loro incompatibili e di incerto futuro.

Nel 1984 Richard Stallman, uno dei personaggi più attivi nello sviluppo del software per Unix, decise di lasciare il laboratorio di Artificial Intelligence del MIT e iniziare lo sviluppo di un nuovo sistema operativo compatibile con lo Unix ma libero dai controlli dell'AT&T; conìò per primo il termine "free software", inteso in contrapposizione al software proprietario, e scrisse il Manifesto GNU ⁽⁷⁾ nel quale venivano affermati i diritti fondamentali per l'utente:

- utilizzare liberamente i programmi software per qualsiasi scopo;
- studiare il funzionamento dei programmi e adattarli alle proprie necessità;
- ridistribuire liberamente le copie;
- migliorare i programmi e rilasciare pubblicamente le modifiche al fine di arricchire l'intera comunità degli utenti.

Negli anni '80 e nei primi anni '90, l'Open Source software continuò i suoi sviluppi; USENET ⁽⁸⁾ e Internet furono di grande aiuto per il coordinamento e il consolidamento di comunità transnazionali di utenti. In particolare due nuovi fattori crearono una situazione favorevole per un ulteriore allargamento della base di programmatori Open Source:

- la diffusione dei personal computer a basso costo e alte prestazioni in tecnologia Intel che metteva alla portata di molti un sistema in grado di supportare ambienti di sviluppo e connessioni a Internet a costi accessibili grazie alla crescita dell'industria degli "Internet Providers";
- l'invenzione del *World Wide Web* ⁽⁹⁾ destinata a incidere profondamente sull'ICT.

Ed è in tale contesto che nel 1991 in Finlandia, Linus Torvalds, uno studente di informatica, spinto dalla curiosità di verificare la possibilità di far convergere un sistema operativo con le caratteristiche di Unix su piattaforma Intel, iniziò a sviluppare la prima versione di Linux. Nelle sue intenzioni Linux doveva essere un sistema operativo aderente ai principi dello Unix ma completamente nuovo e soprattutto privo di qualsiasi componente sulla quale l'AT&T potesse reclamare diritti di proprietà intellettuale. La sua decisione di renderne pubblici i sorgenti via Internet raccolse l'interesse di molti programmatori che iniziarono a collaborare con Torvalds per estendere le funzionalità del nuovo sistema, renderlo stabile, arricchirlo degli strumenti di gestione già sviluppati dalle comunità che si richiamavano al movimento di Stallman (tutti gli sviluppi già realizzati nell'ambito del progetto GNU confluirono rapidamente in ambiente Linux).

La diffusione di Internet al di fuori dei circuiti accademici e specialistici ha determinato negli anni successivi un crescente consenso nei confronti del fenomeno Open Source fino a farlo diventare un'alternativa reale ai sistemi proprietari, a metterlo in competizione, in alcuni casi, con i principali leader di mercato.

aveva già realizzato un gran numero di applicazioni per tale ambiente. Lo sviluppo del "BSD Unix" fu finanziato principalmente dai contratti DARPA (Defense Advanced Research Projects Agency), e un numeroso *network* internazionale di programmatori cooperarono alle attività di manutenzione ed evoluzione di tale sistema. Per molto tempo il "BSD Unix" venne ridistribuito solo ai detentori di una licenza AT&T Unix, solo negli ultimi anni '80, dopo lunghe dispute legali, fu finalmente distribuito sotto la licenza BSD, una delle prime licenze Open Source.

⁷ GNU è un acronimo ricorsivo "GNU is Not Unix" che voleva ribadire la totale indipendenza dall'AT&T.

⁸ USENET è una rete di sistemi, basata inizialmente sullo UUCP (Unix-to-Unix Copy Protocol, protocollo di trasmissione dati a 7 bit per sistemi Unix) e attualmente su Internet, in grado di scambiare messaggi e di riunirli in "newsgroups" relativi a diverse aree di discussione.

⁹ Il WEB, sviluppato da Tim Berners-Lee presso il CERN di Ginevra, fu rilasciato in modalità Open Source.

1.1 Gli aspetti giuridici dell'Open Source

1.1.1 I riferimenti giuridici

Nella dottrina europea il software, sia esso proprietario o *free*, è considerato opera dell'ingegno ed è protetto dalle leggi sul diritto d'autore.

In Italia i principali riferimenti giuridici sono sanciti nella Legge 22 aprile 1941 n.633 "Protezione del diritto d'autore e di altri diritti connessi al suo esercizio", integrata con il decreto legislativo n. 518 del 29 dicembre 1992, emanato in attuazione della direttiva 91/250/CEE, con il quale viene riconosciuto al software il valore di opera dell'ingegno e vengono introdotte alcune disposizioni volte alla sua tutela.

L'esigenza di disporre di strumenti giuridici per la tutela dei programmi per elaboratore ha trovato pieno accoglimento nel nostro ordinamento con la Legge 18 agosto 2000 n.248 "Nuove norme di tutela del diritto d'autore"; sulla base di quanto previsto dall'art. 10 della suddetta legge è stato emanato il Decreto del Presidente del Consiglio dei Ministri 11 luglio 2001 n.338 "Regolamento di esecuzione delle disposizioni relative al contrassegno della Società italiana degli autori e degli editori (S.I.A.E.) di cui all'articolo 181-bis della legge 22 aprile 1941, n. 633, come introdotto dall'articolo 18 agosto 2000, n. 248, recante nuove norme di tutela del diritto d'autore".

Il 22.6.2001 è entrata in vigore la direttiva 2001/29/CE del Parlamento europeo e del Consiglio europeo sull'armonizzazione di taluni aspetti del diritto d'autore e dei diritti connessi nella società dell'informazione.

Secondo la legge italiana l'autore è la sola persona fisica titolare sia dei diritti morali (paternità, inedito, integrità dell'opera) sia dei diritti patrimoniali (riproduzione, diffusione, distribuzione, traduzione, noleggio e prestito) riconosciuti dalla legge. Mentre il diritto morale d'autore è di carattere personale e perciò irrinunciabile e inalienabile, il diritto patrimoniale d'autore consente di trarre dall'opera ogni utilità economica. Nelle forme e nei modi previsti dalle norme del codice civile, i diritti patrimoniali possono, quindi, essere ceduti o concessi in licenza d'uso.

La licenza d'uso di software è una figura negoziale di derivazione nord-americana, che consente l'utilizzazione di un programma a soggetto diverso dal suo originario creatore. Normalmente utilizzata per il software commercializzato in serie, la licenza d'uso può essere definita un contratto con il quale il soggetto (creatore/fornitore) cede a un altro soggetto (licenziatario), di regola dietro corrispettivo, il diritto di utilizzare in modo non esclusivo il programma oggetto di licenza e la relativa documentazione accessoria. Per il software c.d. proprietario, il licenziatario utilizza una versione del programma in forma eseguibile dalla macchina e non invece il codice sorgente.

Il dibattito in dottrina sulla natura giuridica della licenza d'uso è ampio. Un primo orientamento ⁽¹⁰⁾ considera il contratto di licenza d'uso come una vera e propria locazione di bene mobile (software), disciplinata quindi dalle norme del codice civile in materia di locazione (artt. 1571 e 1614 c.c.). Un secondo orientamento ⁽¹¹⁾ pone in evidenza l'assenza di una norma giuridica di riferimento nel nostro ordinamento e sostiene, pertanto, che la licenza d'uso di software debba essere considerata come un contratto innominato (atipico), concluso dalle parti in forza del generale

¹⁰ Galgano, La cultura giuridica italiana di fronte ai problemi informatici, in Alpa-Zeno-Zencovich, I contratti dell'informatica. Profili civilistici, tributari e di bilancio, Milano 1987.

¹¹ Rossello, I contratti dell'informatica nella nuova disciplina del software, Milano 1997.

principio dell'autonomia contrattuale di cui all'art. 1322 del codice civile.

1.1.2 Le licenze Open Source

Normalmente la licenza pone alcune restrizioni in termini di uso, copia, modifica e distribuzione dell'opera, a tutela di un vantaggio economico che ricompensi l'autore del suo lavoro.

Il software Open Source viene rilasciato con particolari licenze che intendono invece rimuovere tali restrizioni e garantire all'utente la libertà di studiare e modificare e ridistribuire l'opera.

Secondo tale principio, nel 1984 R. Stallman definì la GNU Public License (GPL)⁽¹²⁾ quale strumento di tutela legale dei diritti dell'utente pubblicati nel Manifesto GNU della Free Software Foundation (cfr. supra).

La GNU Public License può essere definita come una sorta di contratto che promuove il "permesso d'autore" ⁽¹³⁾, attraverso una clausola che rende impossibile, dopo aver modificato il software, legare il risultato con una licenza proprietaria. In altri termini la clausola è volta a mantenere la libertà stessa del software perché attribuisce a chiunque il permesso di eseguire, copiare, modificare il programma e distribuirne versioni modificate, con l'unico obbligo di trasmettere a tutti i prodotti derivati la licenza GPL.

Di fatto i termini della licenza GPL non sono stati condivisi da tutti i partecipanti alle comunità Open Source e, nel tempo, sono state elaborate svariate licenze che si differenziano principalmente nel grado di libertà concessa all'utente nell'utilizzo del software ⁽¹⁴⁾. Tra le licenze ritenute più permissive vi è la Berkeley Software Distribution License (BSD) che consente, a chiunque, di licenziare diversamente il software BSD e i programmi da questo derivati, con l'unico obbligo di citazione dell'autore originario.

Sia la GPL sia le altre licenze Open Source sono in ogni caso regolate e tutelate dalla normativa sul diritto d'autore che riconosce a quest'ultimo la scelta delle modalità con le quali porre in circolazione la propria opera.

¹² Il testo integrale della GPL è consultabile all'indirizzo www.gnu.org/licenses/gpl.txt

¹³ La locuzione "permesso d'autore" deriva da un gioco di parole ottenuto trasformando "copyright" in "copyleft". Il termine "copyleft", utilizzato di frequente come sinonimo di GPL, è stato coniato per sottolineare la contrapposizione di "right" cui viene, secondo le comunità Open Source, associato un significato eccessivamente restrittivo, al termine contrario "left".

¹⁴ Tale diversità risponde a due diverse visioni del ruolo dell'Open Source. La prima, rappresentata dal movimento di Stallman (Free Software Foundation), è di carattere ideologico e pone come priorità assoluta la produzione di software libero che mantenga tale caratteristica anche nelle modifiche successive; la seconda, che si riconosce nella Open Source Initiative, è più pragmatica e lascia molte più libertà rispetto alla GPL consentendo sia la convivenza di software Open Source con software proprietario sia l'utilizzo a fini commerciali del software Open Source; lo scopo principale per tale movimento è quello di produrre software di qualità e soprattutto di favorirne la distribuzione e l'utilizzo. Le comunità Open Source afferenti ai due diversi movimenti hanno, però, concordato su una formulazione univoca - l'Open Source Definition (OSD) - per sancire i diritti che una licenza software deve garantire per poter essere certificata come licenza 'Open Source'. Nell'allegato B) vengono riportati i criteri stabiliti dalla Open Source Definition. Le licenze Open Source certificate OSD alla data del Giugno 2001 sono circa 20. Tra le più diffuse troviamo: GNU General Public License (GPL), GNU Library or "Lesser" Public License (LGPL), BSD License, MIT License, Artistic License, Mozilla Public License (MPL), Qt Public License (QPL), IBM Public License, MITRE Collaborative Virtual Workspace License (CVW License), Ricoh Source Code Public License, Python, zlib/libpng License, Apache Software License, Sun Internet Standards Source License (SISSL), Intel Open Source License, Jabber Open Source License, Nokia Open Source License, Nethack General Public License. La licenza GPL, con la quale vengono distribuiti circa l'80% dei prodotti e la licenza BSD con una percentuale di impiego intorno al 6%, sono le più utilizzate.

1.2 L'organizzazione e il modello di sviluppo dei progetti Open Source

Gli elementi d'interesse proposti dall'Open Source Software non sono solo di carattere tecnologico, ma riguardano anche profili di natura organizzativa, sociologica ed economica che per taluni aspetti divergono dalle tradizionali regole che governano l'industria del software. Tali aspetti verranno sinteticamente di seguito trattati al fine di fornire una chiave d'interpretazione più ampia del fenomeno e di poterne, quindi, giudicare la reale forza innovativa nel complessivo panorama dell'ICT.

Descrivere il processo che regola la nascita e l'evoluzione di un progetto Open Source è compito reso complesso dalla dinamicità con la quale mutano le categorie di partecipanti ai gruppi di progetto, la quantità e la tipologia delle risorse tecniche disponibili, le fonti di finanziamento.

Infatti, se fino a pochi anni fa la collaborazione a gruppi Open Source era di fatto limitata a organizzazioni scientifiche e singoli individui, il recente interesse di aziende commerciali – sia quelle consolidate quali la IBM, HP, etc. che quelle costituite con l'obiettivo di commercializzare prodotti Open Source e servizi connessi quali Red Hat, SUSE, VA Software – ha di fatto portato nelle comunità di sviluppo elementi di novità sia in termini di nuove categorie di partecipanti che di progetti che favoriscano il *business* della società partecipante.

Un progetto Open Source nasce, di solito, da un'idea maturata da un singolo individuo o da un ristretto gruppo di persone⁽¹⁵⁾. Le ragioni sono diverse e possono essere ricercate in un interesse scientifico, nell'esigenza di disporre di nuovi strumenti per procedere più efficacemente nel proprio lavoro o nella volontà di creare un prodotto funzionalmente equivalente a un prodotto proprietario.

La fase iniziale è quindi condotta da una o più persone coadiuvate da colleghi o conoscenti coinvolti in virtù di una comunanza di interessi sulle medesime tematiche. Il gruppo lavora al progetto fino a che non siano raggiunti risultati presentabili pubblicamente. Tali risultati vengono, quindi, pubblicati su siti *ftp* o *http* dedicati e nel contempo gli obiettivi del progetto vengono resi noti attraverso *mailing list*, *newsgroup*, servizi di *news* in Internet, manifestazioni pubbliche quali seminari o conferenze. Il tutto è spesso ospitato e finanziato da istituzioni scientifiche o da altri tipi di organizzazioni interessate al progetto. Le aziende, soprattutto le maggiori, entrano in un secondo momento del processo, a livello di finanziamenti, quando è chiara una potenziale validità commerciale del prodotto. Questo intervento, di solito, coincide con una fase avanzata dello sviluppo del prodotto che, quindi, già gode di una certa stabilità e maturità.

La diffusione dei risultati dello sviluppo iniziale (codice sorgente del software, documentazione, finalità del progetto, standard di riferimento, etc.) tramite Internet dà il via a un fenomeno di interessamento e collaborazione da parte di terzi che vanno a infoltire il gruppo originario. Il contributo dei nuovi partecipanti è fondamentale per gli elementi di novità che apporta in termini di analisi, approfondimenti, critiche, prospettive alternative sugli obiettivi del progetto e sulle strategie da adottare per garantirne lo sviluppo. In questo modo nuove informazioni e nuove risorse sono continuamente integrate nel processo di sviluppo. I continui rilasci dei risultati e della documentazione nonché lo scambio di informazioni e le discussioni *on line*, instaurano un circolo virtuoso che rappresenta il progetto Open Source. Si è così costituita una comunità che potrà portare o meno a un prodotto fruibile e vitale. Ciò dipenderà dalla validità del progetto, dall'interesse che potrà suscitare, dall'organizzazione e dalla credibilità dei suoi leader.

I partecipanti a un progetto Open Source sono riconducibili principalmente alle categorie degli studenti universitari, dei ricercatori appartenenti a enti di ricerca pubblici e privati, dei programmatori

¹⁵ Esempi tipici sono rappresentati da Linux e dal Web Server Apache iniziato nei laboratori della NCSA e poi rilevato nel 1996 da alcuni utenti che formarono la comunità Apache, che ancora oggi mantiene e aggiorna il prodotto.

interni ad aziende che utilizzano quel particolare software Open Source nel loro lavoro, dei programmatori indipendenti che traggono guadagno dall'aumento di visibilità che gli conferisce la partecipazione e, infine, dei programmatori regolarmente assunti da aziende commerciali che intravedono nello specifico progetto una possibile fonte di utili.

L'associazione spontanea a una comunità di sviluppo Open Source non deve lasciar intendere che non vi siano regole alle quali i partecipanti debbano attenersi per poter collaborare. Esistono numerosi documenti che regolano sia il corretto comportamento delle risorse umane sia il corretto utilizzo delle risorse tecniche (www.gnu.org).

La struttura organizzativa di un progetto Open Source non prevede alcuna gerarchia formale con poteri organizzativi e dispositivi. Accade piuttosto che per un processo di auto selezione chi è interessato alle tematiche del progetto o ritiene di avere le capacità per trattarne con sufficiente competenza un certo aspetto si autocandida per partecipare. Tuttavia i progetti Open Source non sono sistemi anarchici in quanto vi è sempre un *leader* riconosciuto. In alcuni casi è la persona che per prima ha avuto l'idea. E' questo il caso di Linus Torvalds che continua a detenere un potere pressoché totale sull'evoluzione di Linux in termini di accettazione o rifiuto sulle modifiche da apportare al sistema (*benevolent dictator*). In altri casi un ristretto numero di persone opera in qualità di comitato cui è delegato il compito di mantenere lo sviluppo organico del progetto.

Le capacità dei *leader* sono di fondamentale importanza in un contesto che vive grazie a una comunità di partecipanti che collaborano volontariamente e spesso senza una diretta remunerazione economica. L'autorità del capo progetto deve essere sempre dimostrata da argomenti tecnici convincenti; il processo decisionale e l'organizzazione devono essere basati sul merito; le informazioni scambiate tra i vari partecipanti devono essere conosciute e condivise a tutti i livelli della comunità di sviluppo. Laddove la *leadership* del capo progetto non è giudicata idonea a garantire il successo del progetto viene rilevata da altra persona appartenente alla comunità.

In termini pratici ogni progetto prevede alcuni ruoli riconosciuti:

- **Project Manager.** Coordina le varie fasi dello sviluppo, determina lo stato di una release del progetto (pianificazione, pre-alpha, alpha, beta, stabile, maturo), indica le scadenze, decide nel caso ci siano divergenze;
- **Maintainer.** E' il responsabile per uno specifico componente (¹⁶) del progetto e si occupa di tutto ciò che riguarda tale componente. E' l'interfaccia tra il componente e il progetto;
- **Amministratore.** E' responsabile dell'amministrazione delle risorse tecniche quali il sistema hardware e software di supporto e i servizi a disposizione del progetto;
- **Programmatore.** Sviluppa software o lavora alla documentazione per un particolare componente;
- **Utente attivo.** E' colui che, non potendo o non volendo lavorare al progetto, si rende utile in qualche modo segnalando errori o suggerendo nuove caratteristiche o miglioramenti.

¹⁶ Va ricordato che una delle caratteristiche fondamentali del software Open Source è la completa aderenza ai concetti di modularità e riusabilità del software. Grazie alla modularità, ovvero alla suddivisione del lavoro di sviluppo in entità elementari e autoconsistenti, il lavoro condotto in modo distribuito diviene possibile, si riduce la curva di apprendimento richiesta per la partecipazione, si permette a singoli di concentrarsi su una specifica funzionalità, si evitano sovrapposizioni di sviluppo e, infine, si può operare concretamente anche in assenza di coordinamento diretto e senza necessità di frequente interazione.

Una stessa persona può rivestire ruoli diversi così come più persone possono ricoprire lo stesso ruolo. I ruoli, soprattutto quelli che danno maggiore visibilità, sono sempre contendibili.

Ogni progetto cita, nel sito ove risiede, tutti i partecipanti alle attività. Il diffuso pregiudizio secondo il quale chiunque possa modificare il software Open Source è da ritenersi errato poiché solo dopo aver dimostrato competenza e aver contribuito fattivamente si entra a far parte a pieno titolo di una comunità di sviluppo. Guadagnarsi visibilità non è semplice, richiede dedizione, capacità e disponibilità a interagire con gli altri componenti della comunità, tempo da dedicare alle richieste di chiarimenti che giungono numerose attraverso le *mailing list*. Recenti studi sulla composizione delle comunità di progetto Open Source calcolano che per ogni persona che contribuisce a scrivere codice ve ne sono cinque che si limitano a segnalare errori. Questo, tra l'altro, può essere considerato una conferma delle intense attività di verifica cui è sottoposto il software Open Source.

I progetti Open Source attualmente attivi sono alcune decine di migliaia; i maggiori tra questi dispongono di risorse proprie perché finanziati da Università, enti di ricerca o società commerciali e hanno un sito dedicato (¹⁷).

I progetti che non dispongono di risorse proprie vengono ospitati gratuitamente da alcune associazioni che mettono loro a disposizione i servizi Internet, l'hardware, il software e gli strumenti tecnici necessari al progetto. Tra le associazioni che, alla fine del 2002, ospitavano un numero considerevole di progetti, si richiamano (¹⁸):

- Source Forge (www.sourceforge.net), con 52.291 progetti e 525.960 utenti registrati, che rende disponibili le risorse hardware, gli strumenti software, gli archivi e i servizi Internet necessari allo sviluppo dei progetti ospitati. A ciascun progetto è riservata una sezione che contiene le informazioni sul team di sviluppo, le versioni disponibili e il relativo stato, la documentazione, i problemi aperti, la storia delle modifiche e consente il *download* del software;
- The GNU Project (www.gnu.org), con 766 progetti e 6.331 utenti registrati. Gli strumenti tecnici messi a disposizione da questo sito sono gli stessi dell'ambiente Source Forge. Rappresentando uno dei siti storici dell'Open Source, dà grande enfasi alle informazioni e ai documenti che espongono le regole tecniche per l'ordinato sviluppo dei progetti e i suggerimenti di carattere etico e sociale utili a coinvolgere e rispettare gli altri partecipanti;
- Open Source Development Lab (www.osdl.org) è un centro di calcolo sito a Portland (Oregon) sostenuto finanziariamente da 21 aziende sponsor (¹⁹). I laboratori OSD offrono ai programmatori le risorse tecniche per lo sviluppo di progetti che arricchiscano Linux e in generale l'Open Source di funzionalità specifiche per i sistemi di classe Data Center e TELCO. L'iniziativa ha l'obiettivo di ampliare gli sviluppi verso classi di sistemi che non essendo largamente disponibili non riescono a guadagnare l'attenzione delle comunità.

In sintesi si calcola che i partecipanti attivi alle comunità di sviluppo fossero, alla fine del 2002, circa 250.000.

¹⁷ www.kernel.org, www.metalab.uc.edu, www.apache.org, www.samba.org, www.mysql.org, www.sendmail.org.

¹⁸ Le rilevazioni sono state effettuate in data 9 dicembre 2002.

¹⁹ Alcatel, Caldera, Cisco, Computer Associates, Dell, Ericsson, Fujitsu, Hitachi, HP, IBM, Intel, LinuxCare, Miracle, Mitsubishi, MontaVista, NEC, Nokia, Red Hat, SuSE, Toshiba, VA Software.

1.3 Le motivazioni dei partecipanti alle comunità Open Source

Volendo analizzare il movimento Open Source non si può trascurare un punto di fondamentale importanza ovvero le motivazioni che sostengono il lavoro dei partecipanti alle comunità di sviluppo.

Gli studi pubblicati su tale argomento sono numerosi. Alcuni si soffermano principalmente sugli aspetti volontaristici e sociali che contraddistinguono le comunità Open Source, altri danno del fenomeno una visione più pragmatica.

L'intero movimento dell'Open Source è sicuramente permeato da forti ragioni ideologiche che fanno ritenere la diffusione libera e gratuita del software un dovere etico a favore delle categorie sociali più deboli e dei paesi in via di sviluppo. E', infatti, diffusa tra i partecipanti alle comunità Open Source la convinzione che il *digital divide* si possa superare abbassando drasticamente il costo di acquisizione del software e divulgando quanto più possibile ogni informazione possa essere utile alla crescita culturale e tecnologica dei paesi più deboli.

Altri studiosi ⁽²⁰⁾ ritengono tuttavia più corretto ricondurre il movimento Open Source a una particolare forma di attività di ricerca, considerate le forti analogie rilevabili tra gli ambienti scientifici e le comunità Open Source. Ciò in considerazione del fatto che la gran parte della scienza dell'informazione e delle tecniche di programmazione hanno avuto inizio negli ambienti accademici e scientifici e che tali ambienti continuano a sostenere molti dei più importanti sviluppi Open Source.

Gli elementi comuni tra i modelli di sviluppo scientifico e quelli di sviluppo del software Open Source citati dai sostenitori di questa tesi sono:

- la metodologia di lavoro. In entrambi i contesti sono presenti :
 - la consuetudine a scambiare liberamente informazioni al fine di promuovere nuovi progressi nel campo della conoscenza, attraverso un processo di verifica pubblica;
 - la condivisione degli strumenti tecnici;
 - la forte valorizzazione della creatività individuale;
 - la capacità di creare comunità informali legate da comuni interessi scientifici o tecnologici ⁽²¹⁾;
- la contemporanea presenza di forme di competizione e collaborazione. Come le comunità scientifiche anche le comunità Open Source si sviluppano e operano in un clima in cui si esprimono, da un lato, forti livelli di collaborazione tra i partecipanti a uno stesso progetto e tra progetti diversi e, dall'altro, competizione per acquisire credibilità e ruoli di maggiore responsabilità nonché per attrarre nuovi partecipanti e nuove risorse al proprio progetto;
- le fonti di finanziamento. I progetti Open Source attingono, di solito, alle stesse fonti cui attingono le Università e gli enti di ricerca.

²⁰ Nikolai Bezroukov "Open Source Software development as a special type of academic research" (http://firstmonday.org/issues/issue4_10/bezroukov/index.html) e Christopher M. Kelty "Free Software/Free Science" articolo presentato all'International Institute of Infonomics (Maastricht) http://firstmonday.org/issues/issue6_12/kelty/index.html

²¹ Bezroukov, op. citata, sottolinea come le comunità Open Source, operando attraverso Internet, ricordino le comunità informali degli scienziati che in altre epoche storiche condividevano via lettera i progressi raggiunti, le idee e le critiche e come le regole non scritte di tali comunità, ancora valide negli attuali ambienti scientifici, siano, a parere degli storici della scienza, il vero motivo dell'eccezionale sviluppo scientifico e tecnologico del mondo occidentale.

Bezroukov nota, inoltre, come sia le comunità scientifiche sia quelle Open Source siano costantemente attraversate da dispute di carattere tecnico e ideologico (²²) e come in entrambe l'appartenenza a una comunità già influente e l'accesso ai maggiori centri di sviluppo possa drasticamente ridurre la difficoltà di far accettare una nuova idea.

Se si ritiene valida l'analogia tra gli ambienti scientifici e quelli dell'Open Source si può affermare che la motivazione principale a partecipare al lavoro di sviluppo non è direttamente correlata a guadagni economici ma al soddisfacimento di bisogni di livello superiore quali lo sviluppo professionale, il riconoscimento, l'autostima e la reputazione.

Le comunità Open Source mutuano, quindi, le classiche forme di riconoscimento sociale valide nel mondo scientifico quali l'eponimia (Ingo scheduler, Reiser File System, etc.), la citazione (tutti i partecipanti allo sviluppo di un progetto Open Source vengono citati in un'apposita sezione della documentazione che accompagna il progetto), i premi per il miglior prodotto dell'anno conferito nel corso dei convegni.

Come detto in premessa, l'interesse destato dall'Open Source in ambiti commerciali fa sì che altre forze, oggi, contribuiscano al suo sviluppo.

Le aziende distributrici del sistema Linux (Red Hat, SuSE, Caldera, etc.) partecipano, di norma, ai progetti Open Source e la loro motivazione è certamente di carattere economico; il loro *business* si concentra, piuttosto che sulla vendita del software, sull'offerta dei servizi a esso connessi: la personalizzazione del software, l'assistenza sistemistica e il supporto. L'incontro degli interessi commerciali di tali aziende e il mondo dell'Open Source sta operando quale acceleratore nel processo di diffusione dei prodotti. Con l'obiettivo di accrescere la base degli utenti, le attività di tali aziende sono fortemente orientate allo sviluppo delle funzionalità che rendono l'installazione e l'utilizzo dei prodotti alla portata degli utenti non specializzati; collaborano, quindi, con le comunità Open Source svolgendo quella parte di lavoro (definita nel gergo delle comunità "*non sexy*") che difficilmente riesce a trovare programmatori che vogliono occuparsene, non richiedendo particolari competenze tecniche, o che ne abbiano le capacità (scrivere un'interfaccia utente gradevole richiede principalmente conoscenza degli aspetti legati all'ergonomia e alla psicologia).

Le grandi aziende del settore I.T. (IBM, HP, Sun, etc.), che hanno annunciato un forte impegno nei confronti del mondo dell'Open Source, con l'obiettivo di estendere i volumi di vendita dei prodotti complementari al software (hardware e servizi in primo luogo), contribuiscono direttamente agli sviluppi ai quali sono interessati facendo confluire verso le comunità Open Source risorse umane, economiche e tecniche.

1.4 La cultura della condivisione delle informazioni

L'Open Source Software e il lavoro delle comunità di sviluppo sono una delle manifestazioni di una forte e diffusa tendenza culturale che considera la condivisione e la circolazione delle idee uno dei principali motori del progresso scientifico.

²² Il progetto KDE (desktop grafico funzionalmente simile a Windows) è stato a lungo osteggiato da alcune comunità di sviluppo in quanto contenente librerie non Open Source. Altro caso rilevante è stato quello del progetto Samba che, per disaccordi interni alla comunità, ha prodotto il fenomeno del "forking", ovvero la divisione di una comunità in due comunità distinte che lavorano entrambe sulla medesima tematica ma con obiettivi e priorità diversi. A volte, come nel caso Samba, i due filoni riconfluiscono in uno stesso sviluppo guadagnandone benefici in ordine al numero di funzionalità disponibili che risultano essere la somma delle funzionalità sviluppate autonomamente dalle due comunità, in altri casi uno dei due progetti, il più debole in termini di leadership e di componenti, viene abbandonato.

Esempi concreti di tale tendenza si manifestano in vari campi:

- il MIT (*Massachusetts Institute of Technology*) ha iniziato un esperimento di Open Courseware (<http://ocw.mit.edu/index.html>) mettendo gratuitamente a disposizione corsi autodidattici che vanno dall'architettura all'ingegneria, dalle scienze umane e sociali all'arte. Per ogni corso ci saranno anche note di lettura, simulazioni e esami anche se non verranno rilasciati diplomi. La dichiarazione di Charles M. Vest, presidente del MIT, esprime bene lo spirito di tale cultura:

"L'iniziativa OpenCourseWare può apparire contro-intuitiva in un mondo regolato dalle leggi del mercato. Va contro la natura degli attuali valori materiali. Ma è realmente coerente con ciò che considero il "meglio" del MIT. E' innovativa. Esprime la nostra convinzione sul modo in cui l'istruzione può essere migliorata, attraverso il costante ampliamento dell'accesso alle informazioni e stimolando gli altri a partecipare";

- la prestigiosa rivista Nature (www.nature.com) pubblica sul suo sito la rivista completa per gli abbonati e utenti registrati;
- esiste un vasto movimento di opinione che afferma che le informazioni riguardanti la ricerca sul genoma umano sono proprietà dell'intera umanità e non possono essere detenute da una singola organizzazione o nazione (<http://www.ornl.gov/hgmis/elsi/elsi.html>).

2 Classificazione dei prodotti open source e loro diffusione

Le comunità Open Source hanno prodotto una grande quantità di software del quale è tuttora difficile reperire dati certi sulla reale presenza nel mercato. Ciò è dovuto alla molteplicità dei canali attraverso i quali tali software vengono distribuiti, al diverso livello d'interesse suscitato presso le aziende commerciali e la stampa specializzata e, infine, alla tipologia del software (²³). E' infatti largamente riconosciuto che l'attenzione verso il mondo del software Open Source è stata stimolata, da un lato, dalla diffusione di Internet, che vive tuttora di protocolli e servizi Open Source e, dall'altro, dal personale tecnico più giovane che ha conosciuto l'Open Source durante il periodo scolastico e ha, quindi, maturato competenza principalmente sulle componenti sistemistiche di base.

Tale situazione ha creato di fatto una enorme mole di informazioni su alcune categorie di prodotti mentre per altri i dati disponibili sono solo indirettamente valutabili (numero di *download*, notizie riportate dalla stampa, sondaggi, etc.). Tale evidenza consente di individuare un criterio di classificazione del software che si ritiene utile riportare, anche per il prosieguo dello studio.

Le fonti esaminate per classificare il software e darne dati qualitativi e quantitativi sono:

- i siti *web* dove vengono ospitati i progetti Open Source (cfr. sopra);
- i siti *web* (²⁴) che mantengono cataloghi aggiornati dei prodotti suddivisi per materia;
- i dati pubblicati dai fornitori e dai distributori di Linux;
- i dati pubblicati dagli analisti;
- i dati pubblicati dalla stampa specializzata.

Dall'analisi dalle fonti citate emerge con chiarezza che i progetti più consolidati, con un numero consistente di sviluppatori, utenti attivi, partecipanti alle *mailing list* e ai *newsgroup*, con la maggiore dinamicità di modifiche e richieste di informazioni sono quelli che riguardano nell'ordine le seguenti categorie di software:

- il software di base e infrastrutturale;
- il middleware;
- le applicazioni di office;
- il software applicativo;

vengono citati, anche se non rivestono un interesse specifico nel settore bancario, due altri filoni di ricerca molto attivi seppure circoscritti a sviluppi specialistici:

- Linux *embedded*;
- il software di "*frontiera*".

²³ Acquisire dati attendibili sulla diffusione del software della specie presenta svariate difficoltà. Infatti, non sono significative le informazioni sul numero di copie vendute dai fornitori in quanto si tratta di software spesso scaricato da Internet; inoltre, il numero dei *download* non risulta sempre significativo perché non si ha la ragionevole certezza che questo corrisponda al numero d'installazioni. A ciò si aggiunga che le indagini sono prevalentemente orientate all'analisi dei siti IP attivi su Internet, che rappresentano circa il 10% del mercato globale dei server, poiché a questi vanno aggiunti i server collocati sulle reti Intranet aziendali e, in generale, su LAN interne. Per i motivi suddetti, i dati forniti anche da importanti centri di ricerca e di analisi hanno scarsi riferimenti alla situazione reale e forniscono cifre sovente discordanti.

²⁴ Freshmeat (<http://www.freshmeat.net/>) e Linuxapps (<http://www.linuxapps.com/>)

Per ciascuna delle famiglie di software si riportano i prodotti che si sono già affermati o che hanno suscitato un evidente interesse presso gli utenti. Questa sintesi non va considerata un giudizio sulla qualità dei prodotti, non essendo possibile, data la loro numerosità, condurre un'analisi esaustiva.

2.1 Il software di base e il software infrastrutturale

In tale area sono compresi i sistemi operativi, i protocolli e i servizi di rete, i linguaggi di programmazione, i compilatori, i prodotti per la sicurezza, i servizi di clustering, il supporto al calcolo distribuito, l'High Performance Computing, i prodotti software per la realizzazione di servizi Internet/Intranet.

Sistemi operativi:

⇒ **Linux.** E' il sistema operativo scritto da Linus Torvalds e dalla più numerosa comunità di sviluppo Open Source. Ha tutte le funzionalità dei più noti sistemi Unix e ha raggiunto un elevato livello di stabilità, affidabilità e ricchezza funzionale, in particolare nelle componenti di gestione dei servizi e dei protocolli di rete. Linux, sviluppato inizialmente sulle piattaforme Intel a 32-bit, oggi è disponibile su tutte le piattaforme hardware più diffuse (²⁵) comprese le architetture a 64-bit. Il progetto Linux è ospitato, sin dai suoi primi rilasci su Internet nel 1991, dal sito *web* dell'Università del North Carolina (<http://metalab.unc.edu>). Tale sito è quindi una fonte attendibile per tutto ciò che riguarda la composizione della comunità Linux e le sue attività. I dati quantitativi sulla diffusione di Linux sono fortemente contrastanti in quanto i classici strumenti di valutazione per fatturato prodotto o per unità consegnate non possono applicarsi ai prodotti Open Source. Gli analisti di Gartner Group affermano, in una inchiesta condotta dalla divisione Dataquest, che la quota di Linux nel mercato dei server negli USA a fine 2000 era all'incirca dell'8,6%, di cui il 93% distribuito da Red Hat.

Le ricerche pubblicate da IDC (Febbraio 2001) assegnavano ai vari sistemi operativi le seguenti quote percentuali calcolate su un totale di 6,1 milioni di nuovi server consegnati nel corso del 2000:

- 41% Windows
- 27% Linux
- 14% Unix
- 17% Netware
- 2% Altri

Gli analisti concordano su stime di elevata crescita di Linux in ambiente server. La società Meta Group, in un recente studio (²⁶), prevede che Linux nel 2006/7 sarà il sistema operativo installato sul 45% dei nuovi server.

Limitatamente ai server attivi sulla rete Internet una fonte attendibile può essere considerata la Società Netcraft (www.netcraft.com) che rileva, mediante una particolare tecnica d'interrogazione

²⁵ Compaq Alpha, Sun SPARC e UltraSPARC, Motorola 68000, PowerPC, PowerPC64, ARM, Hitachi SuperH, IBM S/390, MIPS, HP PA-RISC, Intel IA-64 più tutta una serie di processori destinati a mercati di nicchia.

²⁶ "META Group's current analysis and five-year forecast for Linux and its impact on IT landscape", 9.12.2002

basata sullo scambio di pacchetti TCP/IP (*header fields*), indicazioni sul sistema operativo dei server attivi e li pubblica mensilmente. La tabella seguente si riferisce alla rilevazione di Settembre 2001:

OS group	% (March)	% (June)	Composition
Windows	49.2%	49.6%	Windows 2000, NT4, NT3, Windows 95, Windows 98
[GNU/]Linux	28.5%	29.6%	[GNU/]Linux
Solaris	7.6%	7.1%	Solaris 2, Solaris 7, Solaris 8
BSD	6.3%	6.1%	BSDI BSD/OS, FreeBSD, NetBSD, OpenBSD
Other Unix	2.4%	2.2%	AIX, Compaq Tru64, HP-UX, IRIX, SCO Unix, SunOS 4 and others
Other non-Unix	2.5%	2.4%	MacOS, NetWare, proprietary IBM OSs
Unknown	3.6%	3.0%	not identified by Netcraft operating system detector

⇒ **FreeBSD** è una versione dello Unix di Berkeley libero da qualsiasi componente proprietario AT&T;

⇒ **selinux (Security-Enhanced Linux)** è una versione di Linux, rilasciata dalla National Security Agency (²⁷) americana, arricchita da meccanismi di sicurezza "forti".

Servizi Internet/Intranet:

⇒ **Apache** (www.apache.org) è il diretto discendente del primo web server sviluppato dal National Center for Supercomputing Applications (agenzia governativa degli USA per lo studio delle applicazioni di supercomputing). Quando, nel 1996, alcuni sviluppatori di NCSA decisero di fondare la società Netscape, gli utenti del web server NCSA formarono la comunità "The Apache Group" che, partendo dal codice sorgente di NCSA, continuò a migliorare il prodotto, aggiornarlo e ampliarne le funzionalità.

Apache è ora un prodotto molto flessibile, configurabile e personalizzabile con l'aggiunta di moduli scritti ad hoc utilizzando le API che lo stesso mette a disposizione; è fornito gratuitamente corredato dal codice sorgente e senza alcuna restrizione d'uso sulle piattaforme Linux, Windows NT/9x, Netware 5.x, OS/2 e sulle principali versioni di Unix.

Dalle rilevazioni di Netcraft, realizzate nell'ottobre 2002, su oltre 35 milioni di siti, emerge che Apache è il *web server* più utilizzato su Internet;

<i>web server</i>	October 2002	%
Apache	10509138	65.39
Microsoft IIS	4431875	25.06
Iplanet	278775	1.42
Zeus	182918	1.35

²⁷ La National Security Agency (www.nsa.gov) è un organismo governativo degli Stati Uniti il cui compito è quello di coordinare, dirigere ed eseguire attività altamente specializzate per proteggere i sistemi informativi statunitensi.

⇒ **Sendmail** (www.sendmail.org) è il mail transfer agent (MTA) più utilizzato su Internet. E' considerato lo standard di riferimento per i servizi di e-mail. Ha funzionalità avanzate di sicurezza (crittografia, autenticazione e il supporto per i filtri antivirus e spam), offre accesso via *web browser*, Wireless Application Protocol (WAP) e apparati i-Mode.

L'elevato livello di configurabilità è il maggior pregio di Sendmail, ma nello stesso tempo il suo maggior "difetto", comportando una notevole complessità soprattutto nelle attività installazione.

Dalle rilevazioni di Gartner (²⁸), Sendmail risulta essere il sistema di spedizione di posta più diffuso essendo utilizzato su oltre il 75% degli Internet Mail server;

⇒ **Postfix** (<http://www.postfix.org>) è un altro server di mail sviluppato da Wietse Venema, uno dei maggiori esperti di sicurezza informatica, presso i laboratori T. J. Watson Research Lab (²⁹) della IBM. Postfix è rilasciato dalla Società IBM con il nome di Secure Mailer;

⇒ **Qmail** (<http://cr.vp.to/qmail.html>) è anch'esso un server di mail simile a Sendmail, disegnato sulla base di precisi principi di sicurezza. Il suo utilizzo è in rapida crescita; secondo gli autori, nell'Ottobre del 2001, più di 700.000 server utilizzavano Qmail quale SMTP server. Tra i più famosi : Yahoo!, NetZero, Address.com, Onelist.com e Critical Path. Il premio promesso nel 1997 dagli autori a chi riuscisse a pubblicare una vulnerabilità attraverso la quale violare Qmail non è stato ancora reclamato;

⇒ **SQUID** (www.squid-cache.org) è un server proxy Open Source ad alte prestazioni che supporta i protocolli Http, Https, Ftp e Gopher. Il prodotto fornisce *caching* e richieste di inoltro per oggetti Internet (pagine *web*, file ftp, etc.). I *web browser* possono utilizzare il *cache server* Squid locale come proxy server HTTP, riducendo il tempo di accesso e il consumo di banda. Squid mantiene gli oggetti in memoria o su disco locale; i server Squid possono essere installati in gerarchie, in modo da consentire ai server centrali di costruire capienti cache di dati disponibili per i server di più basso livello. Il prodotto fornito con licenza GPL è stato concepito per poter funzionare su tutte le piattaforme Unix ed è disponibile anche per l'ambiente Windows NT. Non sono noti dati quantitativi sulla sua diffusione;

⇒ **BIND** (www.bind.org) (Berkeley Internet Name Domain) è l'applicativo di riferimento per quanto riguarda la gestione e l'implementazione del protocollo DNS (Domain Name Server) che ha per compito quello di tradurre gli indirizzi numerici di Internet in una forma mnemonica e comprensibile per tutti. Bind è stato creato presso l'università di Berkeley grazie a un programma finanziato dalla US Defense Advanced Research Projects Administration (DARPA) e viene distribuito gratuitamente dall'Internet Software Consortium (ISC). Bind risulta essere utilizzato dal 95% dei DNS (³⁰).

Prodotti e protocolli di sicurezza

Gli strumenti di sicurezza Open Source sono molto numerosi e ampiamente utilizzati. Molte delle applicazioni e dei servizi commerciali che operano in questo settore utilizzano sviluppi realizzati in ambiente Open Source.

²⁸ Gartner Strategic Analysis Report R-08-3046 J. Browning, J. Graff, L. Latham 26 May 1999.

²⁹ Wietse Venema è un famoso esperto di sicurezza autore di fondamentali strumenti quali "TCP Wrapper" e "The Coroner's Toolkit".

³⁰ www.tecnoteca.it/.../contenuti/SezioniRubriche/20020317185619867068/20020317192805828958/ShowRubriche1 visitato (1/8/2002)

E' praticamente impossibile fornire un elenco di tutti gli strumenti, i protocolli e le applicazioni Open Source disponibili per tale comparto. Di seguito vengono elencate le aree in cui i prodotti Open Source sono maggiormente presenti e consolidati e sono brevemente richiamati i prodotti più noti.

- Protocolli che meritano particolare menzione in quanto diffusamente utilizzati per applicazioni che trattano il trasferimento sicuro dei dati (E-Commerce, etc.) :
 - **OpenSSH** (<http://www.openssh.com/>) è un programma che consente l'accesso a un sistema remoto e l'esecuzione di comandi in modo sicuro, crittografando il traffico tra i sistemi collegati in rete;
 - **OpenSSL** (<http://www.openssl.org>) è una realizzazione in forma Open Source dei protocolli SSL/TLS (Secure Socket Layer e Transport Layer Security) per la certificazione e la comunicazione cifrata. Inizialmente il progetto si chiamava SSLeay, ma da quando l'autore originario lo ha dovuto interrompere, questo è stato ripreso da un gruppo indipendente che lo ha ribattezzato OpenSSL. Il progetto fornisce funzioni per implementare SSL v2/v3 e TLS v1; funzioni di crittografia generale simmetrica, asimmetrica, gestione certificati e CA (Certification Authority);
 - **IPSEC Standard** sviluppato dall'IEFT (Internet Engineering Task Force) **IPSec Working Group**, si tratta di un set di estensioni al protocollo IP che hanno come obiettivo quello di fornire delle funzioni di sicurezza, quali l'autenticazione e la crittografia. **IPSec** offre un servizio simile a SSL ma lavorando a livello network, per essere totalmente trasparente alle applicazioni, è composto da tre protocolli principali: **AH (Authentication Header)** che garantisce l'autenticazione di un pacchetto cifrando con un algoritmo forte l'header IP del pacchetto stesso; **ESP (Encapsulating Security Payload)** che assicura la riservatezza e l'integrità di un pacchetto cifrandone il contenuto (payload = campo dati); **IKE (Internet Key Exchange)** che consente la negoziazione dei parametri di connessione;
 - **Stunnel** (<http://www.stunnel.org>) è un programma che consente di crittografare qualsiasi collegamento TCP all'interno del protocollo SSL (Secure Sockets Layer) disponibile sulle piattaforme Unix, Linux e Windows. Stunnel è utilizzato per crittografare quei protocolli di comunicazione che non supportano SSL (POP, IMAP, NNTP, LDAP, etc).
- Prodotti per la crittografia (GPG, BestCrypt, Transparent Cryptographic File System):
 - **GNU Privacy Guard** (www.gnupg.org) è un sistema di crittografia dei dati derivato dal completo rifacimento Open Source del prodotto crittografico PGP. Non utilizza l'algoritmo brevettato IDEA e può quindi essere utilizzato senza alcuna restrizione. GnuPG aderisce allo standard RFC2440 (OpenPGP);
 - **BestCrypt** (www.jetico.sci.fi) permette di creare dischi virtuali protetti sui quali è possibile memorizzare i dati in forma crittografica e non accessibile;
 - **Transparent Cryptographic File Systems** (www.tcfs.it) implementa un file system tradizionale in cui l'utente ha la possibilità di proteggere i suoi dati in maniera trasparente rispetto alle applicazioni esterne.
- Prodotti per la rilevazione delle vulnerabilità note e in generale delle debolezze nell'impianto di sicurezza dei sistemi e delle reti:
 - **Nmap** è uno scanner di rete che consente di individuare eventuali "porte" liberamente

accessibili presenti sui diversi sistemi collegati in rete;

- **Sara** (Security auditor's research assistant) e **Nessus** (www.insicure.org) sono prodotti per lo scanning dei sistemi; Nessus è ritenuto dagli esperti il miglior prodotto nella sua categoria;
- **Abacus Project** (www.psionic.com/abacus) è una suite di strumenti che fornisce sicurezza agli host attraverso le sue funzioni di rilevamento delle intrusioni;
- **StackGuard** (www.immunix.org) è un compilatore in grado di intercettare e bloccare l'esecuzione di programmi che tentano di attaccare il sistema, per assicurare che non possa essere utilizzato l'attacco basato sul buffer overflow dei programmi eseguibili creati dagli utenti;
- **Wietse's tools and papers** (www.porcupine.org/wietse) è una raccolta di programmi per la sicurezza dei sistemi;
- **Snort** è uno dei più diffusi sistemi di Ids (Intrusion detection system).

➤ Prodotti per l'auditing:

- **SNARE** (System iNtrusion Analysis and Reporting Environment) è programma Open Source che controlla gli accessi al sistema mettendo in evidenza eventuali intrusioni; è composto da tre sottosistemi: un kernel, un componente di auditing e un componente grafico. Il prodotto è disponibile con licenza GPL;

➤ Prodotti per la configurazione sicura dei sistemi:

- **Bastille-Linux** (<http://www.bastille-linux.org>) è un insieme di script che hanno come obiettivo quello di rafforzare la sicurezza del sistema operativo Linux. Attualmente supporta le distribuzioni Red Hat, Debian e Mandrake ed anche HP-UX. Il prodotto è disponibile con licenza GPL.

High Performance Computing:

- **OpenMosix** (<http://openmosix.sourceforge.net>) è una estensione dei kernel Linux che permette la creazione di cluster scalabili ad alte prestazioni, necessarie sia per il calcolo parallelo che per quello sequenziale. OpenMosix è in grado di gestire dinamicamente le risorse di un cluster (cpu, memoria) e di fornire, allo stesso tempo, un'interfaccia trasparente agli utenti e alle applicazioni. Con tale prodotto l'utente può eseguire un certo numero di processi di calcolo da un'unica macchina, essendo cura del sistema distribuire questi processi tra le varie macchine per bilanciare il carico di lavoro all'interno del cluster. Il software viene distribuito con licenza GPL;
- **Beowulf** (<http://www.beowulf.org>) è un cluster di PC a elevate prestazioni per il calcolo parallelo massivo costruito attraverso l'interconnessione, su rete privata ad alta velocità, di componenti hardware di uso comune (PC) sui quali sono in esecuzione sistemi operativi Open Source (Linux e OpenBSD). Beowulf nacque nell'estate del '94 quando due ricercatori del CESDIS (Center of Excellence in Space Data and Information Sciences), Thomas Sterling e Don Becker, costruirono un cluster basato su 16 processori Intel 486-DX4 connessi da un canale Ethernet. Il sistema ebbe un successo immediato all'interno della comunità accademica e della NASA. Oggi un "Beowulf-class parallel computer" - un cluster di 64 PC con Linux connessi da LAN privata - è operativo presso il

GSFC (NASA - Goddard Space Flight Center) per il progetto theHIVE (Highly-parallel Integrated Virtual Environment). L'obiettivo di theHIVE è mostrare le potenzialità di un sistema per il calcolo parallelo a basso costo e la sua capacità di soddisfare le esigenze di un ambiente complesso. Utilizzato principalmente in ambito scientifico è di recente oggetto di studio anche presso alcune banche per applicazioni di *risk management*.

Strumenti di sviluppo:

La disponibilità di tool di sviluppo Open Source, anche molto sofisticati, è ampia. Comprende i compilatori per i linguaggi di programmazione più diffusi: **GCC** (GNU C Compiler), **gforth**, **FreePascal** e ambienti per lo sviluppo di applicazioni *web*: **Perl** (persino più diffuso di Java), **Php**, **Python**, **Tcl**. Particolare interesse si è concentrato di recente sul progetto **Eclipse** (www.eclipse.org) avviato dalla società IBM che ha stimato in circa 40 milioni di dollari gli investimenti già sostenuti su tale software e di recente donato alla omonima comunità di sviluppo. Si tratta di una piattaforma di sviluppo Open Source che ha l'obiettivo di integrare in un unico ambiente i tool di sviluppo, gli ambienti di test e i *repository* provenienti da vari produttori. Il framework è basato su un sistema di API aperte e *plug-in* che permette a qualsiasi terza parte di sviluppare un tool specifico che sarà, poi, utilizzabile direttamente all'interno dell'IDE (Integrated Development Environment). Partecipano al progetto le società Merant, che produce i pacchetti PVCS per la gestione della configurazione, Qnx Software Systems (Qssl), leader nel settore dei sistemi operativi in tempo reale, TogetherSoft, produttrice di Together ControlCenter, la prima piattaforma per il Model-Build-Deploy e Rational, produttrice dell'omonima piattaforma per lo sviluppo di software. Sun Microsystems sostiene invece una iniziativa analoga con NetBeans, che ha raccolto un certo consenso tra i produttori di sistemi Unix. Attualmente il progetto Eclipse rimane una risorsa per sviluppatori Java, anche se sono in via di completamento versioni C e C++. Al progetto - che funziona con sistemi Linux, Windows e Unix - partecipano più di 1200 sviluppatori di 63 paesi, la componente *framework* conta circa 2 milioni di *download*.

2.2 Middleware.

A tale categoria di software si riferiscono i prodotti che, oltre a fornire servizi elaborativi specializzati (Data Base Management System, application server, etc.), consentono alle applicazioni di interagire con il software di base e i protocolli di rete attraverso estese interfacce di programmazione.

⇒ **MySQL** (www.mysql.com) è un database relazionale Open Source che gode di buona reputazione tecnica. Sta registrando un periodo di grande crescita con una base dichiarata di 3 milioni di utenti tra i quali alcune grandi aziende: Yahoo!, MP3.com (Vivendi Universal), Motorola, NASA, Silicon Graphics, HP, Xerox, Cisco e Texas Instruments. Nel sito ufficiale del progetto sono reperibili le informazioni sulle applicazioni realizzate da tali aziende con l'utilizzo di MySQL.

Prove prestazionali e di scalabilità condotte dai tecnici delle riviste eWeek e PC Magazine, pubblicate nel 2001, hanno messo a confronto MySQL con i più importanti prodotti proprietari (Oracle, DB2, SQL Server). I risultati di tali prove pongono MySQL in posizione confrontabile, e per alcuni aspetti superiore, rispetto ai DBMS proprietari (fonte: eWeek del 25/2/2002).

I diritti sul codice sorgente del prodotto, così come il marchio MySQL, sono di proprietà della Società svedese MySQL AB.

Il prodotto è disponibile sia con licenza GPL sia con una licenza commerciale. La scelta di quest'ultima è destinata a quegli utenti/aziende che, intendendo utilizzare il codice MySQL all'interno dei loro prodotti proprietari, non vogliono essere vincolati alla distribuzione dei loro software con

licenza GPL. La licenza commerciale è, quindi, preferita soprattutto da quelle aziende che producono *appliance* o apparati elettronici dedicati a una specifica funzione realizzata con software *embedded* che utilizza al suo interno MySQL.

⇒ **SAMBA** (www.samba.org). E' un'applicazione che consente a server Linux/Unix di offrire a client Microsoft Windows le funzioni di condivisione di files e stampanti, di autenticazione degli utenti nonché di instaurare relazioni di *trust* con server Windows NT e, nel prossimo futuro, con Windows 2000. Non sono noti dati sulla sua diffusione ma è considerato un prodotto strategico dai maggiori fornitori Unix che partecipano alla comunità di sviluppo, con personale proprio in modo da avere maggiori garanzie sull'evoluzione del progetto.

⇒ **Gli Application Server**

- **JBoss** è un application server Open Source basato sullo standard J2EE. E' completamente sviluppato in Java ed è uno dei progetti più vitali nell'intero panorama dell'Open Source (nel sito che lo ospita vengono dichiarati circa 100.000 *download* mensili); seppure non si hanno informazioni dirette sul suo utilizzo, si registra presso la stampa specializzata un forte interesse e una grande attenzione sui suoi possibili utilizzi;
- **Tomcat** è un'implementazione Open Source delle tecnologie Java Servlet e Java Server Pages sviluppata dall'Apache Software Foundation;
- **Zope** è un application server e un content management system completo di strumenti di sviluppo e database per gli oggetti. E' utilizzato tra gli altri dalla Comunità Europea, dalla Marina degli Stati Uniti, dalla General Electric Aircraft Engines, dalla NASA, dal Dipartimento dell'agricoltura degli Stati Uniti e da alcune università ⁽³¹⁾.

2.3 Le applicazioni di *office*

La disponibilità di applicazioni Open Source per l'ambiente client è molto vasta e comprende applicazioni per la produttività personale, la messaggistica, l'accesso al mondo Internet e la gestione di contenuti multimediali. In generale le applicazioni Open Source in tale comparto supportano una più ampia gamma di piattaforme e di configurazioni internazionali rispetto alle loro controparti commerciali.

Suite Office

➤ **OpenOffice** (www.openoffice.org) è un progetto Open Source che ha per scopo lo sviluppo di un ambiente di prodotti per l'ufficio, basato sullo standard aperto XML per il salvataggio dei documenti; è disponibile in lingua italiana e in altre 25 lingue. La suite è multiplatforma: è disponibile infatti per Windows, Solaris e, ovviamente, Linux. Inoltre sono ormai a buon punto gli adattamenti per i sistemi operativi FreeBSD, Mac OS X e IRIX. Le licenze in uso da OpenOffice sono la LGPL (Lesser General Public License) e la SISSL (Sun Industry Standards Source License). OpenOffice v.1.0 ha la stessa base di codice sorgente presente nel prodotto StarOffice 6.0 di Sun Microsystems ed è, come lo stesso StarOffice, un prodotto di *Office* dalle funzionalità complete. Una delle caratteristiche di OpenOffice è la compatibilità con i prodotti di

³¹ Maggiori informazioni sull'utilizzo di Zope e sulle applicazioni realizzate sono consultabili nel sito www.zope.com nella sezione "case studies".

Office della Microsoft: è in grado di gestire documenti MS Word, fogli di calcolo MS Excel e presentazioni create con Power Point. OpenOffice è composto da un word processor (**Writer**), uno spreadsheet (**Calc**), un programma per le presentazioni (**Impress**), uno per la grafica vettoriale (**Draw**) e offre un valido editor di pagine HTML. Comprende numerosi strumenti per la creazione di formule matematiche e un correttore ortografico. OpenOffice è considerato uno dei più importanti progetti Open Source; vengono dichiarati circa 7.5 milioni di linee di codice e attualmente conta più di 4.5 milioni di *downloads*;

- **KOffice** è la suite integrata di applicazioni per office automation contenuta all'interno della distribuzione KDE ⁽³²⁾ (www.kde.org). Comprende un word processor (**KWord**), un programma di presentazione (**KPresenter**), un foglio di calcolo (**KSpread**), un programma per la generazione dei report (**Kugar**) e un programma per creare istogrammi o grafici a torta (**KChart**). KOffice può essere considerato una valida alternativa per i piccoli uffici o per ambienti domestici;
- **AbiWord** è un editor di testi completo, simile a Microsoft Word. Il programma è adatto per lettere, memo, desktop publishing. Ottimo sostituto di WordPad, supporta formattazione del testo, allineamento giustificato, stili personalizzati, multicolonne etc.. L'obiettivo di questo progetto è quello di sviluppare una suite completa per l'ufficio Open Source in grado di funzionare su qualsiasi piattaforma. AbiWord è in grado di importare ed esportare quasi tutti i formati più diffusi compresi i documenti WORD, RTF ed HTML. L'aspetto più interessante di questo programma è il formato nativo dei suoi files che hanno estensione *.abw e sono scritti in XML, quindi in formato ASCII. Per questo motivo i files di AbiWord possono essere importati da qualunque editor di testi conosciuto;
- **LaTeX** è un vero e proprio linguaggio di programmazione che consente di gestire il testo e le immagini nel modo desiderato. I suoi campi di applicazione vanno dai memorandum alle lettere commerciali o personali, dalla stesura di articoli e pubblicazioni scientifiche, ipertesti, *slides*, fino alla realizzazione di libri;
- **Evolution** è un Personal Information Manager, progettato e realizzato dalla Ximian (<http://www.ximian.com>) per l'ambiente grafico GNOME, con l'obiettivo di creare una vera e propria alternativa Open Source a programmi come Outlook di Microsoft. Evolution include tutte le funzionalità standard di gestione della posta elettronica e delle mailbox nonché filtri definiti dall'utente e ricerche rapide. Supporta un calendario flessibile e consente agli utenti di creare e confermare incontri di gruppo online. Evolution è un tool di gestione delle informazioni personali e di gruppo per sistemi Linux e UNIX.

Browser

- **Mozilla** www.mozilla.org è un *browser* nato qualche anno fa quando Netscape decise di rilasciare i sorgenti di Navigator favorendo così la nascita di un team intenzionato a sviluppare un nuovo *browser* Open Source. Mozilla supporta tutti gli standard attualmente diffusi, permette la navigazione in più finestre separate oppure più schede dentro la stessa finestra, supporta Java, flash ed altri plug-in. Contiene un client di posta, un editor visuale per creare pagine *web*, un gestore di dati personali, un cookie manager, un client IRC, un newsreader, un analizzatore della struttura del documento e altre numerose funzioni. Mozilla è gratuito ed è funzionante su Windows, Linux, Mac OS e sui maggiori sistemi Unix;
- **Konqueror** (<http://www.konqueror.org>) è il *web browser* Open Source incluso nell'ambiente

³² E' un ambiente desktop grafico Open Source per workstation Unix.

grafico KDE in grado di svolgere le funzioni di file manager, di *browser* Internet e di visualizzare documenti e immagini. Diversamente da Mozilla e Netscape è molto più leggero e veloce e la visualizzazione di alcune pagine *web* è simile a Internet Explorer. Konqueror ha tra le sue funzionalità il supporto di Java Applets e Java Scripts.

Emulatori

- **FreeHost3270** (<http://freehost3270.sourceforge.net>) è una soluzione Open Source per l'accesso ad applicazioni "legacy" 3270. FreeHost3270 è una applicazione *web* che distribuisce, in modalità applet Java, su un numero illimitato di client l'emulatore per la connessione a host. Inoltre fornisce delle API (application program interface) per le funzionalità di "Screen Screeing" per l'utilizzo delle interfacce in modalità Html.

Client di Posta

- **Mutt** (<http://www.mutt.org>) è un programma di posta Unix Open Source molto semplice da usare ma potente al tempo stesso. Client di posta in modalità testo, supporta IMAP/POP e NNTP. Viene distribuito con licenza GPL;
- **KMail** (<http://kmail.kde.org>) è il programma di posta elettronica fornito nell'ambiente grafico KDE. Con questo programma è possibile gestire più caselle di posta, le news e tutto ciò che riguarda i servizi gratuiti. Kmail è in grado di fornire un alto grado di personalizzazione anche per quello che riguarda la modalità di visualizzazione dei messaggi.

2.4 Il software applicativo *enterprise*

Rientrano in questa categoria di prodotti software le applicazioni di natura operativo-contabile largamente presenti in ambiti aziendali che coprono esigenze specifiche (applicazioni di contabilità, gestione magazzino, ecc.). Tale software è quello che mostra attualmente minore vitalità. Tra i pochi progetti attivi si segnalano:

- **Erp Compiere** è l'applicazione per le soluzioni di *business* più popolare tra i prodotti Open Source, conta circa 310.000 (³³) *downloads* ed è costantemente presente tra i primi dieci progetti più attivi di Source Forge. Compriere, distribuito con licenza Mozilla Public License (MPL), fornisce una soluzione integrata, in grado di offrire una risposta completa ai bisogni informativi delle aziende di piccole e medie dimensioni: dall'amministrazione e gestione del cliente, sistemi di approvvigionamento, gestione ordini, pianificazione e controllo delle vendite, servizi di distribuzione, gestione della contabilità, Call Center. Inoltre il prodotto ha anche funzionalità di commercio elettronico quali: catalogo on line, transazioni sicure con la banca, riconoscimento clienti e analisi delle richieste *web*. Comprende anche un sistema di workflow per lo svolgimento guidato delle attività con modelli già predisposti per le attività ricorrenti. Il prodotto è disponibile anche in modalità ASP (Application Service Provisioning) rendendo possibile usufruire dei servizi applicativi collegandosi tramite Internet al server.

³³ <http://www.compiere.org/index.html> (visitato il 2/8/2002).

- **GNU Enterprise** è un progetto iniziato dalla Free Software Foundation per lo sviluppo di una suite di applicazioni che offrano funzioni finanziarie dotate del supporto per più nazionalità e più valute nonché strumenti di gestione dei bilanci aziendali, delle attività di commercio elettronico, delle risorse umane, dei progetti, delle supply chain, della produzione e delle vendite. Il progetto è ancora in fase prototipale ma può essere considerato un segnale di allargamento degli interessi delle comunità di sviluppo verso problematiche applicative.

2.5 Linux *embedded*

La disponibilità in forma sorgente di Linux ha reso possibile ed economicamente conveniente la realizzazione di sistemi operativi c.d. *embedded* dedicati a far operare apparati elettronici di uso comune (telefoni cellulari, palmari, televisori e automobili). Si tratta di adattamenti del codice su processori e piattaforme diverse dall' x86 (Arm, Mips, Ppc) e su microcontroller industriali come il Pc/104. Uno dei progetti più interessanti, in questo settore, è Rtlinux (Realtime Linux) attualmente utilizzato su sistemi industriali, nei simulatori di volo, nei veicoli di spostamento subacqueo e dalla Nasa in varie applicazioni proprietarie. Disponibile per processori x86, Ppc ed Alpha, può essere eseguito anche da un floppy e su schede Pc/104 standard. Versioni *embedded* di Linux sono attualmente impiegate per la realizzazione di Point of Sale, router, sistemi specializzati per applicazioni multimediali. Tra gli apparati più noti troviamo l'Internet phone Aplio basato su Voip (Voice over Ip) collegabile ad Internet tramite una Lan o tramite dial-up, il robot Isamu, progettato dall'Università di Tokio, che può salire le scale, e compiere una varietà di altre azioni, il *WebPad* Progear per la navigazione wireless su Internet e dotato di tutte le più comuni funzioni dei PDA e vari altri PDA quali il Nasa Assistant e Hangil che incorpora tutte le funzionalità tipiche di tali apparati (*browser*, lettore di posta elettronica, pdfview, mp3 player, word processing, voice recorder, image viewer). Il mercato del software *embedded* è oggi stimato in circa 23 miliardi di dollari. Ulteriore informazioni sugli sviluppi Open Source attualmente attivi si trovano nel sito <http://www.linuxdevices.com>.

2.6 Il software di "*frontiera*"

Le comunità Open Source hanno oggi allargato la base dei ricercatori impegnati nei settori dello sviluppo del software scientifico e della ricerca avanzata in campo informatico. La facilità di interazione garantita da Internet ha reso possibile la formazione di gruppi di progetto eterogenei i cui confini superano i centri di ricerca pubblici e privati. In questo campo molti dei concetti innovativi e dei paradigmi avanzati vengono sviluppati all'interno di comunità Open Source che vedono la collaborazione di ricercatori e volontari. Alcuni di questi prodotti vengono poi inglobati in progetti commerciali, altri rimangono a livello embrionale mentre altri ancora prosperano ed evolvono come progetti Open Source di successo.

Fra gli esempi di successo di tale paradigma possiamo citare:

- ILIAS (www.ilias.uni-koeln.de) progetto Open Source per l'*e-learning* nato presso l'Università di Colonia e finanziato dal Ministero della Scienza e della Ricerca della regione Renania-Vestfalia. Viene utilizzato dall'Università di Colonia dall'anno 2000 e conta circa 12.000 utenti;
- il campo dell'intelligenza artificiale, sia con progetti su linguaggi e prodotti di base (LISP, PROLOG, reti neurali, genetic algorithms and programming, fuzzy logic), sia con progetti specializzati (robotica, interpretazione semantica e *web* semantico, ricerche su agenti intelligenti, rappresentazione della conoscenza e sistemi esperti, etc.);
- i modelli di elaborazione e collaborazione con particolare riferimento a:

- P2P (*peer to peer*): progetti che realizzano modelli di elaborazione distribuita (SETI, GNUTELLA, JABBER, FREENET, etc.);
 - l'emergente campo del GRID computing che promette di definire un nuovo modello di interconnessione e condivisione delle capacità elaborative utilizzato attualmente per attività di supercomputing. Il principale progetto di sviluppo dell'architettura Grid è Open Source (www.globus.org);
- gli strumenti e i modelli crittografici. La crittografia moderna del resto, riferendosi al principio di Kerkhoff, secondo cui la sicurezza di un sistema crittografico è basata esclusivamente sulla conoscenza della chiave e non sulla segretezza dell'algoritmo di cifratura e decifrazione (³⁴), ben si sposa con il concetto di Open Source.

³⁴ Auguste Kerkhoff "La Cryptographie Militaire", 1883.

3 Le aree di possibile utilizzo con riferimento alle banche

I sistemi informativi delle banche sono costituiti da ambienti tecnologici e applicativi estremamente diversi.

Al fine di poter correttamente evidenziare e valutare le opportunità offerte dai prodotti Open Source si ritiene opportuno considerare diversi scenari.

3.1 Ambiente Mainframe

E' stato tradizionalmente il sistema informatico per eccellenza della banca. L'evoluzione tecnologica degli ultimi anni ne sta ampliando le possibili funzioni consentendone una più facile integrazione con gli ambienti e le architetture distribuite.

Nel seguito sono evidenziati i diversi ruoli e vengono formulate alcune considerazioni.

3.1.1 Mainframe Tradizionale

L'ambiente bancario transazionale non è stato fino ad ora di particolare interesse per la comunità Open Source; ciò va ricondotto anche al fatto che gli strumenti e i linguaggi tradizionalmente usati in tale ambiente (COBOL/PL1) sono diversi rispetto a quelli usati prevalentemente nel mondo Open Source (C/Java).

Particolare interessante da evidenziare è il fatto che in quest'area, almeno per i prodotti applicativi installati, i sorgenti del software, anche se non ceduti contrattualmente dal fornitore, sono disponibili per l'adeguamento agli standard aziendali e per effettuare le ricompilazioni nell'ambiente in cui saranno utilizzati. Ciò comporta la possibilità di effettuare interventi di manutenzione correttiva in caso di situazioni di estrema urgenza, impegnandosi con i fornitori a comunicare tempestivamente le rettifiche effettuate.

Relativamente allo scenario evolutivo, la disponibilità dei servizi Unix (Unix system services) apre nuove prospettive per l'integrazione del mondo applicativo Java con l'ambiente transazionale tradizionale. In quest'ambito potrebbero aprirsi spiragli per poter utilizzare software Open Source già disponibile, sia pure limitatamente alla realizzazione di componenti di interesse generale piuttosto che per esigenze specifiche dell'ambiente bancario.

3.1.2 Mainframe Unix

L'evoluzione dei servizi Unix su tecnologia mainframe di fatto oggi fa di tali servizi una piattaforma efficace e proponibile per la realizzazione di applicazioni anche critiche. La facilità di integrazione con gli ambienti tradizionali e la maggiore efficienza del colloquio, che rimane locale, apre la possibilità di introduzione di applicazioni realizzate con tecnologie più nuove anche sugli ambienti tradizionali.

La disponibilità di application server Java, quale ad esempio WebSphere, dell'ambiente di sviluppo "C" e dei servizi Unix agevola l'individuazione in ambito Open Source di applicazioni e/o utilities da poter utilizzare ed eventualmente far evolvere. Attualmente la disponibilità di prodotti Open Source per tale piattaforma è limitata, in particolar modo per ciò che riguarda le applicazioni tipicamente bancarie, e comporta attività di *porting* per adattare i sorgenti allo specifico sistema operativo.

3.1.3 Mainframe Linux

Il sistema operativo Linux è disponibile in ambiente mainframe come partizione completamente separata dall'ambiente OS/390 e quindi dai servizi Unix. Questa implementazione offre alcune soluzioni tecnologiche innovative. Per esempio, utilizza appieno gli strumenti di interconnessione veloce (HiperSockets) da e verso le altre partizioni zSeries, z/OS compreso.

Per questa famiglia di prodotti valgono le stesse regole commerciali, di distribuzione, di supporto e manutenzione applicate agli altri prodotti Open Source, quindi possono essere liberamente prelevabili da Internet oppure ci si può rivolgere a un distributore.

In generale, gli aggiornamenti sono facilmente reperibili sulla rete e le politiche di aggiornamento sono quelle della comunità Open Source. La manutenzione può essere curata sia da IBM che da altri fornitori di servizi fra i quali la SuSE che, tra i distributori di Linux, è stata la prima a rilasciarne una versione per OS/390. Va precisato che, ad oggi, per eseguire più di 15 partizioni su zSeries è necessario disporre di uno specifico software aggiuntivo, z/VM di proprietà IBM.

La disponibilità di Linux in tale ambiente apre la strada al potenziale utilizzo di qualsiasi prodotto Open Source disponibile. Essendo un contesto nuovo vengono di seguito proposti alcuni casi in cui tale soluzione può essere presa in considerazione:

- si rende necessario consolidare in un unico ambiente elaborativo un elevato numero di server. Questo tipo di scelta diventa economicamente interessante quando il numero di server da considerare sia elevato e questi siano omogenei in termini di funzionalità. Altri benefici immediatamente riscontrabili potrebbero essere: la riduzione dei costi gestionali, i risparmi sulle infrastrutture di rete (le diverse realtà infatti colloquiano tramite i canali interni della macchina) e i costi di ambiente (questi ultimi in realtà potrebbero diventare meno percepibili con l'introduzione dei nuovi "rack" server ad alta densità). Ulteriore risparmio, conseguente al consolidamento, in questo contesto potrebbe essere quello relativo alle risorse umane perché z/VM rende disponibili sofisticati strumenti di monitoring e gestione dei *virtual server* che consentono la manutenzione di molte istanze in modo automatico; non vanno infine sottovalutati i vantaggi in termini di affidabilità garantiti dalle architetture hardware e la possibilità di far fronte velocemente a una crescente domanda di potenza elaborativa;
- vi sono risorse non impiegate appieno su mainframe; in tal caso si potrebbero accogliere le proposte commerciali IBM per valersi di nuovi ambienti applicativi a costo molto basso e senza dover acquisire sistemi dipartimentali dedicati;
- si sceglie di ampliare la disponibilità di applicazioni in ambiente mainframe con il rilascio di applicazioni *mission critical* architetturealmente innovative sullo stesso hardware dove sono conservati i dati *enterprise*.

La strategia di IBM di investire su piattaforme aperte anche negli ambienti mainframe offre una buona garanzia sul futuro del prodotto; la qualità del supporto ne garantisce l'adozione anche presso aziende che non hanno grandi competenze tecniche all'interno.

Va, infine, ricordato che l'opportunità di arrivare a gestire una sola versione delle applicazioni su tutte le piattaforme hardware, da un lato riduce le necessità di *porting* e quindi i costi di manutenzione del software sia per il fornitore che per gli utenti, dall'altro garantisce la massima interoperabilità tra i diversi ambienti, facilitando quindi eventuali decisioni in caso di necessità di sostituzione dell'infrastruttura.

I punti di attenzione sui quali soffermarsi prima di compiere una tale scelta sono principalmente connessi: ai numerosi elementi di novità proposti da tale soluzione e pertanto alla difficoltà di valutarne l'ambito di utilizzo; ai costi di realizzazione e, infine, alla competitività

dell'emergente tecnologia dei "*blade server*" ad alta densità che promette di offrire a costi ridotti un'equivalente disponibilità dei sistemi.

Di seguito vengono riportati alcuni tra i casi di adozione di Linux su mainframe (³⁵).

Banca Popolare di Milano. Ha realizzato il progetto multicanale (Internet Banking, agenzie, Corporate banking, promotori finanziari) in ambiente Linux su Z/VM.

Shenzhen Development Bank. Ha realizzato servizi di Internet banking utilizzando un mainframe IBM S/390 TurboLinux (distribuzione in lingua cinese di Linux);

Deutsche Bank. Utilizza Linux e WebSphere su mainframe per fornire l'accesso via *browser* alle applicazioni legacy in ambiente CICS;

Security Industry Automation Corporation, Salomon Smith Barney, Northern Trust. La realizzazione di soluzioni *e-business* fortemente integrate con l'ambiente *legacy* e l'apertura verso ambienti di sviluppo innovativi J2EE sono le principali motivazioni alla base di tali scelte.

3.2 Ambienti distribuiti

Gli ambienti distribuiti (o dipartimentali) nell'architettura generale dei sistemi informativi aziendali, grazie anche alla diversificata offerta di mercato, si configurano come realtà estremamente variegata sia per tecnologia che per utilizzi applicativi.

Al centro dei sistemi distribuiti ci sono i server, una categoria di apparati molto ampia e diversificata sia in termini di caratteristiche tecniche e di piattaforme operative che in termini d'incidenza dei costi sostenuti per le licenze del software e dei servizi professionali a esso correlati. I costi per le licenze del software sono una componente maggiore nei server di fascia medio-bassa e tendono a divenire una componente minore sui server di maggiori dimensioni (secondo uno studio della McKinsey in una soluzione ERP il 30% dei costi è riferito alle licenze del software mentre il restante 70% è rappresentato dai costi dei servizi di consulenza, installazione, configurazione, etc.).

Di seguito ci si riferirà a tre classi distinte di apparati e per ciascuna verranno esaminati i possibili ambiti di utilizzo del software Open Source.

3.2.1 Server infrastrutturali

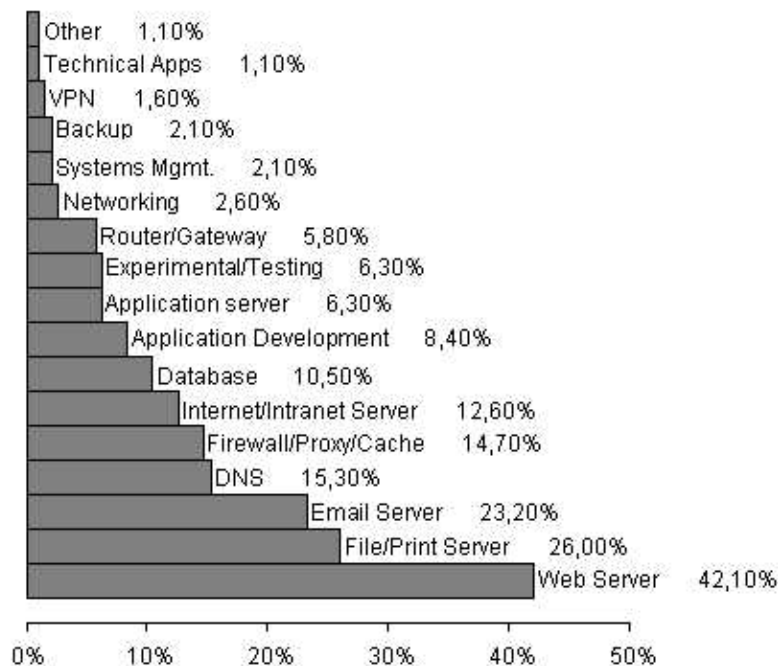
Sono sistemi di fascia medio-bassa dedicati a funzioni specifiche di gestione/connesione alle reti per i quali è richiesta la scalabilità orizzontale e la condivisione del carico elaborativo attraverso meccanismi di *load balancing*. Operano generalmente quali:

- network server: firewall, Domain Name server, DHCP server, proxy e caching server;
- server Internet/Intranet: *web* server, Mail server, Fax server, Telex server, File e Printer server, sistemi per la realizzazione di Virtual Private Network.

I sistemi operativi più utilizzati sono: Unix, Windows, Linux. Va rilevato che in tale segmento di mercato vi è una forte presenza dei c.d. *server appliance*, apparati specializzati per il singolo servizio, già configurati e pronti per essere connessi alla rete. Linux e in generale i prodotti Open Source sono diffusamente utilizzati dai fornitori all'interno di tali apparati, ma configurandosi questi come "*utility*" la scelta non riguarda le piattaforme quanto piuttosto aspetti quali la semplicità operativa, i ridotti tempi di installazione, la qualità tecnica del prodotto.

Il grafico sotto riportato indica le percentuali di impiego di server Linux, sul totale dell'installato, suddivise per servizio erogato (fonte IDC, 2000).

³⁵ I casi di adozione di Linux su mainframe sono stimati dalla società IBM in alcune centinaia e in costante crescita.



I prodotti Open Source sono quelli trattati nella categoria del software infrastrutturale nel capitolo precedente; l'offerta in tale comparto è, quindi, vasta, di buona qualità, multipiattaforma (la quasi totalità dei prodotti è disponibile per i più diffusi sistemi operativi) e comprende prodotti leader di mercato.

Tra i prodotti commerciali, con funzioni equivalenti, disponibili in ambiente Linux, si possono citare WebSphere della Società IBM e SunONE della Società Sun Microsystems.

Tali dati confermano la validità, dal punto di vista tecnico, dell'Open Source nel comparto dei server infrastrutturali.

I casi di successo sono numerosi, come anche testimoniato dall'elevato numero di *web* server Apache su piattaforma Linux attivi sulla rete Internet.

3.2.2 Server applicativi

Operano quali intermediari tra i client e i sistemi di *back end* (database server e transaction server) o ospitano applicazioni aziendali. Richiedono un buon livello di scalabilità verticale e funzioni di cluster evolute. I sistemi operativi più utilizzati sono: Unix, Windows 2000, Linux.

Tali sistemi possono operare in qualità di:

- **Application Server**, intendendo con tale termine il software *middleware* che offre tutte le funzionalità per lo sviluppo di applicazioni *web*. Un Application Server normalmente fornisce in modo nativo funzioni di: connessione con database esterni; interazione con gli ambienti transazionali; controllo del flusso informativo tra i diversi livelli elaborativi; gestione della sicurezza; raccolta dei dati statistici degli accessi e diversi strumenti per la realizzazione della logica di *business* cui sono dedicati.
In questo ambito i prodotti Open Source più noti sono Zope, Tomcat e JBoss.

Il mercato degli Application Server comprende una vasta offerta di prodotti commerciali perché i fornitori tendono a integrare in tale middleware tutti i componenti necessari a interagire con i loro prodotti proprietari. I leader del mercato sono: IBM WebSphere, Oracle Application Server, BEA WebLogic, SunONE della Sun Microsystems. Elemento di particolare interesse è la disponibilità di tali prodotti su piattaforma Linux.

- **Server applicativi per le applicazioni enterprise.** In questo settore troviamo principalmente i sistemi ERP e CRM.

I prodotti Open Source (Compiere e GNU Enterprise) appaiono ancora immaturi, ma su essi si registra l'interesse di alcune comunità.

Tra i prodotti commerciali disponibili in ambiente Linux troviamo SAP, SAS e JD Edwards.

- **Server applicativi general purpose** per l'elaborazione di applicazioni bancarie quali l'on-line banking, mobile banking, “applicazioni di canale”, acquisizione/diffusione di informazioni finanziarie e per tutte le applicazioni di uso generale quali sistemi di marketing, Decision Support System, OLAP, Data mining, gestione logistica, etc.

In tale ambito, la disponibilità di prodotti e soluzioni Open Source è ancora ridotta.

Peraltro si segnala la crescente disponibilità di prodotti commerciali su piattaforma Linux. I casi più importanti di adozione di Linux sono:

Crédit Suisse First Boston, che ha migrato la propria applicazione *mission critical* per l'elaborazione di circa 35 milioni di transazioni al giorno da sistemi RISC/Unix a sistemi Intel/Linux;

Merrill Lynch, che, secondo quanto riportato sul sito www.forbes.com, sta procedendo all'installazione di server Linux con l'obiettivo di ridurre i costi e di garantire, per il futuro, la portabilità delle applicazioni su sistemi diversi (mainframe, server, PC, laptop);

Dresdner Kleinwort Benson, che ha adottato un cluster Linux ad alte prestazioni per ridurre i tempi di elaborazione di informazioni finanziarie;

Shenzhen Commercial United Bank che utilizza Linux per attività di *e-commerce*.

3.2.3 I server di back-end

I server di back-end normalmente ospitano dati e applicazioni *mission critical*. Gli elevati livelli di affidabilità richiesti hanno sinora limitato la scelta delle piattaforme hardware e software a un ristretto numero di soluzioni proprietarie.

Il sistema operativo più utilizzato è lo Unix nelle varie versioni sviluppate dai diversi fornitori e adattate sul loro specifico hardware.

Nel comparto dei Data Base Management System (DBMS) i prodotti Open Source più noti sono MySQL, PostGRES, InterBase e SAP DB. Tali prodotti fanno registrare una crescente diffusione sui sistemi di fascia medio-bassa in tecnologia Intel; non vi è ancora evidenza di un loro utilizzo sui sistemi di back-end. E' prevedibile, nel prossimo futuro, un'espansione dell'utilizzo di Linux anche su quest'ultimo comparto, favorita dalla crescente presenza di hardware basato sui nuovi e più potenti processori della società Intel (³⁶).

³⁶ Gli analisti di Meta Group prevedono che entro il 2006/07 il 95% dei nuovi server saranno basati su hardware Intel.

I prodotti commerciali di maggior interesse, disponibili su piattaforma Linux, sono il DBMS della Oracle e il DB2 dell'IBM.

3.2.4 Alcune considerazioni

I criteri che guidano la scelta della piattaforma server sono fortemente influenzati, oltre che dalle valutazioni tecniche, dalle esigenze funzionali da soddisfare, dall'interoperabilità con i sistemi esistenti, dall'organizzazione e dalle competenze del personale interno.

I soli aspetti tecnici non bastano a qualificare una soluzione come "ottima"; è piuttosto il giusto equilibrio tra i diversi fattori funzionali, organizzativi e gestionali che garantisce il successo. In ogni caso, maggiore è la varietà delle soluzioni proposte dal mercato, maggiore è la possibilità per l'utente finale di individuare ciò che meglio si adatta alle proprie esigenze.

Le opportunità offerte dall'Open Source sono, quindi, da ricercarsi principalmente nell'ampliamento delle possibili soluzioni con riguardo a:

- sistema operativo; in tale settore Linux si propone quale alternativa valida su molteplici piattaforme hardware; la sua diffusione, unita alla standardizzazione dell'hardware, potrebbe portare nei prossimi anni alla trasformazione dei server in prodotti "*commodity*"; il beneficio principale di un siffatto processo è quello di spostare l'attenzione sulla scelta delle applicazioni con le quali realizzare i processi di *business* aziendali piuttosto che sulla sottostante piattaforma hardware e software;
- scelta tra un'ampia gamma di prodotti Open Source - di carattere sia infrastrutturale che applicativo ⁽³⁷⁾ - disponibili su tutti i più diffusi sistemi operativi;
- disponibilità di numerosi prodotti commerciali su Linux.

Negli ultimi anni la scalabilità e l'affidabilità dei sistemi tende ad essere assicurata, secondo un modello orizzontale, da più sistemi che, condividendo i dati, concorrono all'elaborazione delle applicazioni; i prodotti Open Source sono allineati a tale tendenza e offrono nella quasi totalità dei casi capacità di *load balancing* e funzioni di *clustering* evolute.

3.3 Postazioni Client

Le postazioni client nel sistema informatico aziendale ospitano normalmente prodotti di *Office* (elaboratore di testi, foglio elettronico, strumenti per la preparazione di presentazioni grafiche), *browser* per l'accesso a Internet/Intranet, posta elettronica, emulatori di terminale, applicazioni CAD evolute, applicazioni commerciali e/o sviluppate internamente.

In questo comparto ormai da molti anni si assiste al predominio dei prodotti della società Microsoft. Le stime degli analisti assegnano alla Microsoft quote di mercato superiori al 90%, mentre un 4% è detenuto dalla società Apple e un altro 4% da Linux (fonte IDC).

Di conseguenza, il mercato delle applicazioni aziendali e domestiche per i *client* si è sviluppato quasi esclusivamente sulle piattaforme Microsoft.

Nonostante ciò le novità in questo comparto sono numerose ⁽³⁸⁾. Tra le più importanti possiamo citare StarOffice ⁽³⁹⁾ - una suite di prodotti per le attività di *Office* della società Sun che

³⁷ Cfr. capitolo 2.

³⁸ Cfr. capitolo 2.

³⁹ StarOffice è disponibile sulle piattaforme Linux, Unix e Windows.

ne ha rilasciato i sorgenti alla comunità di sviluppo Open Office (www.openoffice.org) - e il consolidamento del prodotto Open Source Evolution con funzioni di Personal Information Management equivalenti a Microsoft Outlook.

L'analisi comparativa tra le funzionalità dei prodotti Open Source con gli omologhi in ambiente Windows è stata oggetto di studi, condotti principalmente dalle pubbliche amministrazioni, che hanno messo in evidenza come, anche a fronte di una pressoché allineata compatibilità funzionale, siano prevedibili forti resistenze da parte degli utenti ad adottare i nuovi prodotti, principalmente per la diversa interfaccia grafica, per la necessità di scelta, a carico dell'utente, del formato in cui salvare i dati e per la disomogeneità con i prodotti utilizzati sul Personal Computer domestico.

Ciò nonostante gli analisti concordano sul fatto che gli sviluppi del mercato di Linux nel settore *client* potrebbero riflettere la diffusione di StarOffice e/o Open Office. Le società IBM e Sun hanno confermato tale tendenza.

L'attenzione verso l'Open Source in questo comparto (soprattutto per le aziende di dimensioni significative) è data dal contenimento dei costi; su questi prodotti in genere le necessità di ricorrere a manutenzione e supporto da parte del fornitore sono piuttosto limitate e viceversa i costi di licenza incidono in modo significativo sul budget aziendale.

Allo stato attuale le opportunità offerte dall'Open Source riguardano:

- i sistemi di accesso a server (*thin client*); in questo caso i prodotti necessari all'operatività dell'utente sono gli strumenti di interfaccia quali *browser*, emulatori 3270 o VT100, *client* di posta elettronica; l'adozione di prodotti Open Source, in tal caso, garantisce il contenimento dei costi, senza presentare particolari rischi;
- i sistemi orientati all'*end user computing* che utilizzano prodotti di *Office*; è possibile, in tal caso, valutare la possibilità di adottare OpenOffice sia su piattaforma Windows sia su Linux. Studi specifici sull'utilizzo dei prodotti di *Office* riportano che l'80% degli utenti non utilizza più del 20% delle funzionalità. Pertanto, una ricognizione che individui sia le effettive necessità nell'utilizzo dei prodotti di *Office*, sia le aree più propense ad accettare i cambiamenti, potrebbe offrire il vantaggio di:
 - accrescere la competizione anche nel comparto *client* ⁽⁴⁰⁾;
 - ridurre la spesa complessiva relativa alle licenze;
 - mantenere viva l'attenzione sull'evoluzione di questi prodotti, favorendo la soluzione delle problematiche di interscambio dei documenti e, in generale, di interoperabilità tra le applicazioni.

Di contro, vanno considerati i costi derivanti dalla diversificazione dei posti di lavoro e i rischi legati a possibili operazioni di *lock-in* sui prodotti oggi più diffusi attraverso l'apposizione di brevetti sulle interfacce e sui formati; tale operazione potrebbe rendere impraticabile per la comunità OpenOffice garantire l'allineamento con le nuove versioni. Considerato che i prodotti Open Source di maggior successo e di migliore qualità sono stati, sinora, quelli contraddistinti da un elevato livello di innovazione tecnica, si potrebbe assistere a un graduale abbandono degli sviluppi di OpenOffice;

- i sistemi sui quali operano applicazioni aziendali realizzate in ambiente Microsoft Windows con l'utilizzo di tecnologie specifiche di tale ambiente (DCOM, WinAPI, MFC, etc.). Considerare l'adozione dell'Open Source in questo caso equivale a un cambio di piattaforma, che, a sua volta, richiede la migrazione delle applicazioni. I costi di tale

⁴⁰ Il governo britannico ha dichiarato di aver firmato nel marzo scorso un contratto con la Microsoft che garantisce risparmi dell'ordine di 147 milioni di sterline nei prossimi tre anni (cfr. rivista InfoWorld del 5 giugno 2002).

intervento, proporzionali alla complessità dell'applicazione, potrebbero non giustificare la migrazione, a meno che nella strategia aziendale non siano prevalenti considerazioni di tipo diverso, quali l'affrancamento da un unico fornitore, ovvero l'estensione del ciclo di vita della piattaforma e delle applicazioni che su essa operano.

4 *Le ragioni a favore dell'open source*

I vantaggi connessi con l'adozione di prodotti Open Source, analogamente a quanto avviene per l'utilizzo di ogni nuova tecnologia, dipendono da fattori specifici relativi all'organizzazione e alla strategia informatica dell'azienda. Di seguito vengono riportati i potenziali vantaggi dell'Open Source, fermo restando che i fattori sopra citati possono esercitare la funzione di moltiplicatore o di inibitore di tali vantaggi.

4.1 **Bassi Costi**

Il primo vantaggio associato al software Open Source è costituito dai bassi costi di acquisizione. I prodotti Open Source sono, infatti, distribuiti gratuitamente o a costi "una tantum" molto contenuti (⁴¹).

Tuttavia tali costi sono solo una componente del **Total Cost of Ownership (TCO)** che si può definire come "la somma di tutte le spese e i costi associati all'acquisto e all'uso di apparati, materiali e servizi".

Gli analisti concordano nel valutare in circa il 20% l'incidenza sul TCO dei costi di acquisto delle licenze e dei relativi aggiornamenti.

Un approfondimento sul TCO associato ai prodotti Open Source Software è riportato nel capitolo 7.

4.2 **Aderenza agli standard**

Il contributo positivo degli standard allo sviluppo dell'economia, sia per effetto del miglioramento dei processi industriali che per l'estensione dei mercati, è stato ampiamente trattato dalla letteratura (⁴²). La diffusione di Internet dimostra, in modo esemplare, come l'uso degli standard riesca a creare nuove opportunità di *business* e di lavoro. L'intera rete Internet si basa su standard a essa collegati (TCP/IP per il protocollo di trasporto, SMTP per la posta elettronica, DNS per la risoluzione dei nomi dei domini, etc.) che sono il risultato delle attività di alcuni organi di standardizzazione (⁴³). Durante l'intera fase di definizione di un nuovo standard le specifiche di dettaglio dei protocolli sono rese note e vengono sottoposte a una revisione critica di tipo pubblico.

Le attività di tali organismi prevedono la realizzazione di una versione di riferimento dello standard che viene rilasciata in modalità Open Source con il duplice obiettivo di:

- diffondere e affermare lo standard;
- rendere possibili le verifiche di conformità.

Le modalità di sviluppo del software Open Source sono rese possibili dall'interoperabilità tra gruppi geograficamente distribuiti. L'obiettivo principale dei progetti Open Source è quello di essere "accessibili" al maggior numero di persone e da qualunque piattaforma informatica. Non è interesse delle comunità Open Source creare barriere d'ingresso all'utilizzo di una nuova tecnologia e pertanto non si registrano in tale ambiente attività di *estensione* proprietaria degli standard specificati.

⁴¹ Poche centinaia di € per una distribuzione di Linux comprensiva degli strumenti di sviluppo, di una vasta molteplicità di applicazioni e della documentazione.

⁴² Un interessante studio "Economics benefits of standardization" è stato pubblicato dal DIN German Institute for Standardization.

⁴³ L'ISOC (Internet Society), il World Wide Web Consortium - W3C, L'Internet Engineering Task Force - IETF.

L'aderenza agli standard è di fondamentale importanza per tutti quei soggetti (pubblica amministrazione, sistema bancario, ecc.) che debbono offrire servizi a una platea molto vasta di interlocutori esterni.

4.3 Affidabilità, qualità e adattabilità

La progettazione, la realizzazione e il rilascio del software proprietario - senza rendere disponibili i codici sorgenti - si ispirano a strategie commerciali e di marketing con cui le aziende mirano a creare continuamente nuove fonti di reddito per poter sopportare i costi di ricerca e sviluppo.

Accade talora che, per acquisire vantaggi competitivi, vengano rilasciati prodotti software prima ancora di aver risolto tutti i problemi e averne verificato la stabilità, salvo procedere a una serie di correzioni successive.

Il continuo rilascio di nuovi prodotti e nuove versioni anche quando le effettive novità sono poche, "costringe" gli utenti finali a una estenuante rincorsa che ha impatti sulla stabilità dei sistemi informatici.

Il software Open Source non ha in generale questi problemi perché gli sviluppi seguono obiettivi di natura diversa quali il miglioramento di un prodotto già esistente o la creazione di un nuovo prodotto di cui si avverte l'esigenza. In quest'ottica non c'è interesse a rifare qualcosa che funziona.

Il modello di sviluppo Open Source rende possibile a un gran numero di utenti la verifica del corretto funzionamento del software, cosicché il prodotto finale risulta essere più stabile e affidabile. Lo stato del prodotto Open Source, riportato nel sito che lo ospita, è determinato dal project manager secondo criteri tecnici e non commerciali.

L'indipendenza dalle logiche del mercato fa sì che i prodotti Open Source possano funzionare su classi differenti di piattaforme hardware (es. il sistema operativo GNU/Linux è disponibile, nelle varie versioni, dal palmare, ai sistemi a 32 bit, fino al mainframe IBM S/390) e software (la gran parte del software Open Source è in grado di operare sui più diffusi sistemi operativi).

4.4 Flessibilità

La disponibilità del codice sorgente offre agli utenti diverse possibilità di utilizzo del software Open Source:

- come prodotto software non modificabile, esattamente come un prodotto proprietario, del quale non si hanno a disposizione i codici sorgenti. Ed è questo, ad esempio, il caso di Linux che può essere acquisito tramite un distributore, installato sulle proprie macchine attraverso le procedure guidate di installazione, reso operativo come qualsiasi prodotto proprietario e assistito dal supporto di un fornitore;
- come prodotto software al quale apportare alcune modifiche per estenderne le funzionalità, per farne accettare in input formati dati non previsti o da adattare a una piattaforma software obsoleta, per la quale non è prevista la sostituzione a breve termine;
- come prodotto software cui apportare modifiche sostanziali al fine di renderlo interoperabile con altre componenti senza dover ricorrere alla realizzazione di complessi strati software di integrazione come accade per i prodotti proprietari; tali interventi se di ampia portata possono rendere il software Open Source equiparabile a un prodotto software sviluppato in casa che necessita, pertanto, dei presidi di manutenzione e supporto equivalenti a quelli che l'azienda ha posto in essere per lo sviluppo software.

Indipendentemente dalla scelta circa la modalità di utilizzo del software Open Source, si può ritenere che la flessibilità di tale software offra garanzie di "vitalità" nel tempo dei prodotti.

4.5 Indipendenza dai fornitori

Le modalità di sviluppo del software adottate dalle comunità Open Source e le licenze che tutelano tale software non consentono la nascita di monopoli. La disponibilità del codice sorgente e la diffusa conoscenza delle tecnologie utilizzate nello sviluppo dei prodotti Open Source, quali i linguaggi di programmazione e le tecniche di comunicazione, garantiscono l'affrancamento dalla dipendenza di singoli fornitori, aumentano la forza contrattuale degli utilizzatori finali e, infine, favoriscono la creazione di un mercato del software maggiormente concorrenziale.

4.6 Protezione degli investimenti

Il software proprietario evolve secondo logiche di mercato.

Le nuove versioni dei prodotti proprietari, che poi nel breve/medio termine risultano essere le sole garantite dal supporto del fornitore, vengono molto spesso rilasciate con certificazioni di compatibilità che impongono all'utente l'aggiornamento di altre componenti software e in alcuni casi il rinnovo delle piattaforme hardware per l'aumento di risorse elaborative richiesto dai nuovi prodotti.

L'utilizzo di prodotti Open Source consente di programmare, con maggiore autonomia, l'utilizzo di versioni successive e, al limite, di non utilizzarle affatto con il vantaggio di:

- salvaguardare gli investimenti fatti in termini di hardware e software senza essere esposti alle scelte strategiche del fornitore (es. eliminazione di un prodotto dal catalogo);
- mantenere una maggiore autonomia decisionale nel pianificare gli aggiornamenti.

4.7 Supporto

Il supporto viene talvolta considerato uno dei punti negativi del software Open Source.

Invero, per i prodotti Open Source largamente diffusi (compresi nelle categorie del software di base e infrastrutturale), si ha a disposizione (siti *web*, *newsgroup*) un'ampia documentazione tecnica volta a spiegare le tecniche di installazione, di configurazione e di risoluzione dei problemi. Non va trascurata inoltre l'ampia partecipazione della comunità di sviluppo alla risoluzione dei problemi: è frequente che chi segnala l'inconveniente, disponendo del codice sorgente, provveda alla pubblicazione della correzione. Di notevole interesse è la documentazione raccolta nelle Frequently Asked Question (FAQ) contenente tutte le domande che rivestono interesse generale, con relative risposte da parte della comunità di sviluppo. Normalmente la consultazione delle FAQ è sufficiente a far superare la gran parte delle problematiche tecniche.

A questa tipologia di supporto, si è aggiunta anche quella che fa riferimento alle modalità tradizionali che contraddistinguono il software proprietario; società come IBM o Red Hat sono in grado di stipulare contratti di assistenza del tutto equivalenti a quelli in uso per i prodotti proprietari.

5 Profili critici del software open source

In questo capitolo saranno analizzati i profili critici del software Open Source e i possibili fattori di rischio.

5.1 Modello di *business*

Gli aspetti economici connessi con il carattere gratuito del software Open Source costituiscono uno degli argomenti maggiormente trattati dagli studiosi della materia ⁽⁴⁴⁾, che generalmente trovano appropriata, per tale tipologia di prodotti, la strategia di spostare la fonte dei profitti dalla vendita del software ai servizi a esso connessi.

Uno dei modelli di *business*, legati a tale strategia, conosciuto come "*Branding and distribution*", è quello generalmente adottato dalle società che distribuiscono Linux (Red Hat, Su.S.E. ed altre minori); esso si concentra principalmente nella vendita, a costi molto contenuti, di supporti contenenti Linux, un vasto insieme di prodotti Open Source, una procedura d'installazione molto semplice, la documentazione e il supporto tecnico per un breve periodo. Tali società offrono inoltre, a costi di mercato, servizi di aggiornamento del software, di personalizzazione e di assistenza tecnica dai quali ricavare i profitti. Una variante di tale modello, conosciuta come "*Best Knowledge here*", prevede la fornitura di tutti i servizi professionali che normalmente vengono richiesti sul software (consulenza, sviluppo, integrazione, formazione, supporto sistemistico e certificazione di prodotti). E' questo il modello di *business* che, grazie ai ridotti investimenti economici necessari all'avvio, viene più frequentemente adottato da nuove imprese a carattere locale.

I fornitori di hardware e software, già presenti sul mercato con una loro offerta proprietaria, stanno generalmente perseguendo la strategia del c.d. "*loss leader*" ⁽⁴⁵⁾ che prevede l'uso di software Open Source quale veicolo commerciale per poter mantenere o aumentare una posizione sul mercato per i propri prodotti proprietari. E' una strategia che ha già dimostrato la sua validità in ambiti circoscritti (concedere in via gratuita un prodotto client per agevolare le vendite del corrispondente prodotto server), ma sul lungo periodo e in presenza di un'offerta di prodotti Open Source che si posizionano sempre più frequentemente in concorrenza con i prodotti proprietari potrebbe mostrare i propri limiti.

Fermo restando che la storia ormai trentennale dello sviluppo dell'Open Source ha dimostrato l'indipendenza del lavoro delle comunità dalla presenza di validi modelli di *business* ⁽⁴⁶⁾, è tuttavia uno svantaggio per l'utente finale non disporre di evidenze quantitative che dimostrino la reale sostenibilità economica di tali strategie commerciali. Ciò impedisce la corretta valutazione della solidità finanziaria delle nuove aziende il cui *business* si concentra sui prodotti della specie, e, nel caso dei grandi fornitori, rende difficile comprendere il valore strategico della loro politica commerciale a favore di tali prodotti.

⁴⁴ A. Bonaccorsi, C. Rossi "L'Economia degli Standard e la Diffusione delle Tecnologie. L'Open Source non è un Assurdo Economico", Laboratorio di Economia e Management, Scuola di Studi Superiori Sant'Anna di Pisa; Josh Lerner, Jean Tirole "The simple economics of Open Source", Journal of Industrial Economics, 2001.

⁴⁵ Con questo termine, che tradotto letteralmente significa "protagonista nelle perdite", si intende la strategia commerciale che prevede la distribuzione in perdita di qualcosa di valore nella speranza che l'utente compri prodotti complementari dai quali ricavare profitti (es. l'hardware, i servizi o altro software proprietario).

⁴⁶ I recenti fallimenti di alcune aziende minori distributrici di Linux non hanno avuto contraccolpi sullo sviluppo dello stesso.

5.2 Assenza di responsabilità legale sul software

Sebbene anche il software commerciale venga generalmente rilasciato con licenze d'uso che contengono precise clausole di limitazione di garanzia e di responsabilità per danni diretti o indiretti causati dal software, è possibile, laddove esista la necessità di garanzie specifiche, stipulare contratti ad hoc con il fornitore del software.

Per i prodotti Open Source tali garanzie non possono essere legalmente assicurate in quanto, come già detto in precedenza, sono le comunità di sviluppo che detengono la proprietà intellettuale del software prodotto e ne curano l'evoluzione e la manutenzione.

5.3 Pianificazione e controllo dei progetti

Le modalità di sviluppo di un progetto Open Source non garantiscono che uno specifico sviluppo inizi o termini con un prodotto consolidato e in tempi predefiniti. La nascita di un nuovo progetto, infatti, si concentra inizialmente sulla definizione degli obiettivi funzionali e sulla predisposizione di un prototipo. L'accoglienza che il progetto potrà avere non è facilmente prevedibile così come la sua vitalità nel lungo periodo.

Tale svantaggio può essere minimizzato adottando una politica di selezione dei prodotti che escluda quelli che non abbiano raggiunto versioni stabili.

Nel caso vi sia l'interesse a seguire l'evoluzione di uno specifico progetto, che non ha ancora raggiunto livelli di stabilità, la partecipazione attiva alla comunità di sviluppo offre la possibilità di valutare meglio i tempi e le modalità di realizzazione.

5.4 Difficoltà nell'individuare l'offerta di prodotti

Le informazioni sull'offerta dei prodotti software proprietari raggiungono gli utenti attraverso una serie di canali che vengono attivati dalle aziende fornitrici secondo precise strategie di marketing. La promozione dei nuovi prodotti e delle nuove tecnologie avviene attraverso campagne pubblicitarie, articoli di stampa, brochures, e-mail, presentazioni, seminari tecnici.

Tale processo di promozione non riguarda i prodotti Open Source se non, nei tempi più recenti, quelli di maggior successo (Linux, Apache, etc.). Le informazioni sugli altri prodotti, anche quelli di equivalente validità, sono più difficili da reperire, essendo pubblicate esclusivamente su siti *web* specialistici. La valutazione della rispondenza alle esigenze aziendali, dei costi e dei benefici correlati è, quindi, un'attività complessa e onerosa.

5.5 Disomogeneità dei prodotti software

Il rischio che vengano prodotte versioni disomogenee, e in alcuni casi incompatibili, dello stesso prodotto Open Source è strettamente correlato al vantaggio di poter apportare modifiche a un programma con grande facilità e senza vincoli legali.

Questo rischio può manifestarsi:

- a livello di comunità di sviluppo; come già precedentemente accennato il fenomeno del *forking* di un progetto è uno dei modi in cui vengono risolte controversie tecniche o organizzative all'interno di una comunità;
- tra le numerose distribuzioni di Linux non sempre completamente compatibili tra loro;
- all'interno di un'azienda dove, per risolvere problemi specifici, si può verificare l'utilizzo di versioni diverse dello stesso prodotto Open Source.

Nel primo caso il pericolo è solo potenziale in quanto sino a oggi il *forking* non ha portato alla frammentazione di un prodotto ma allo sviluppo di un prodotto diverso della stessa categoria (è il caso delle due interfacce grafiche oggi maggiormente diffuse su Linux, KDE e GNOME) o alla

riconfluenza dei due progetti verso un unico prodotto.

Il secondo caso è all'attenzione delle comunità Open Source; nel caso Linux vi sono attività per la definizione della Linux Standard Base (LSB) ⁽⁴⁷⁾.

Per fronteggiare casi di proliferazione di versioni diverse dello stesso prodotto in ambito aziendale è necessario organizzare un presidio che abbia il compito di controllare e governare i processi di adozione e personalizzazione dei prodotti.

5.6 Competenze tecniche

L'adozione del software Open Source richiede un'attenta riflessione sulla necessità di acquisire o riorientare le competenze tecniche presenti in azienda per due motivi.

Il primo, di carattere più generale, è strettamente legato alla mancanza di un fornitore che operi quale agente di diffusione di un nuovo prodotto o tecnologia. Per poter, quindi, considerare, nel processo di scelta di un prodotto, la validità dell'offerta in ambito Open Source, è necessario un coinvolgimento attivo da parte del personale interno o sostenere i costi di una Società di consulenza specializzata in tale attività.

Il secondo riveste importanza a seconda delle competenze già presenti in azienda e dell'utilizzo previsto del software: utente finale di un prodotto consolidato, utente con necessità di apportare alcune personalizzazioni al software, utente programmatore che utilizza un prodotto Open Source quale base di partenza per lo sviluppo di un nuovo prodotto. Le grandi organizzazioni, di solito, avendo un sistema informativo dove convivono diverse piattaforme tecniche, possono riutilizzare le competenze acquisite in ambiente Unix, sicuramente più omogenee con il mondo Open Source, sia in termini di architetture che di ambienti di sviluppo. Le realtà minori dove gli investimenti sulle piattaforma proprietarie sono importanti, l'adozione di prodotti Open Source può incidere in modo significativo sulle voci di costo relative alle risorse umane.

5.7 Ridotta disponibilità di applicazioni commerciali

L'offerta di applicazioni commerciali in ambiente Open Source (Linux) è ridotta rispetto a quella disponibile su piattaforme proprietarie, in particolare negli ambienti MS Windows.

Tale situazione sta migliorando molto rapidamente a seguito dell'interesse dei grandi fornitori che sempre più spesso rendono disponibili i loro prodotti proprietari ⁽⁴⁸⁾ anche su piattaforma Open Source. Con l'aumento della base di installazioni Linux, anche i maggiori Independent Software Vendor (ISV) stanno gradualmente rilasciando i loro prodotti su tale piattaforma (Oracle, SAP, etc.). Per ciò che riguarda invece le piccole e medie aziende produttrici di software principalmente indirizzato all'*end user computing*, l'atteggiamento è sicuramente più cauto dati gli alti costi fissi di migrazione del software che si giustificano solo con una base d'installato molto ampia.

Anche la disponibilità di driver per periferiche specializzate (stampanti, scanner, schede video) è ancora ridotta rispetto alle piattaforme Windows.

⁴⁷ Il progetto LSB ha come obiettivo quello di sviluppare e promuovere un insieme di standard ai quali tutte le distribuzioni di Linux devono attenersi per garantire, a livello applicativo, la compatibilità fra le diverse distribuzioni di Linux. E' di buon auspicio il fatto che tra i promotori dell'iniziativa compaiano i protagonisti del mondo Linux.

⁴⁸ L'IBM ha reso disponibili in ambienti Linux DB2, MQSeries, Tivoli, WebSphere, Sun Microsystems la piattaforma SunONE.

6 La sicurezza

Nel dibattito sul tema del software Open Source gli aspetti di sicurezza sono tra i più controversi. È d'altro lato noto che la diffusione di Internet e delle tecnologie di comunicazione a basso costo ha esposto i sistemi informativi a nuove minacce, acuendo la sensibilità nei confronti di tale tema.

A favore di un maggior livello di sicurezza del software Open Source rispetto ai prodotti proprietari vengono addotte, in generale, le seguenti considerazioni:

- la possibilità di conoscere a fondo e documentare i processi elaborativi che trattano i dati più sensibili è un requisito fondamentale in termini di trasparenza e quindi di sicurezza;
- la consapevolezza dell'utente riguardo ai rischi cui sono esposti i sistemi; l'approccio ai temi della sicurezza conosciuto come "*security through obscurity*" viene rifiutato non solo dal movimento Open Source ma anche dai maggiori esperti di sicurezza nonché sancito dal principio di Auguste Kerckhoffs⁽⁴⁹⁾, padre della crittografia moderna; è da ritenersi decisamente più utile l'immediata pubblicazione degli aspetti insicuri dei sistemi e, laddove possibile, proporre correzioni o *work-around*, piuttosto che mantenere segrete tali vulnerabilità almeno sino a che non sia resa disponibile la correzione;
- la disponibilità dei codici sorgenti aumenta la probabilità di individuare le vulnerabilità e di intervenire velocemente per la loro eliminazione⁽⁵⁰⁾;
- le regole sottostanti al modello organizzativo delle comunità di sviluppo dell'Open Source rendono remota l'ipotesi di programmatori malevoli che inseriscano *trapdoor*⁽⁵¹⁾ o, in generale, codice estraneo nei programmi. Nessuno vorrebbe legare il proprio nome a un atto contrario alla filosofia dell'Open Source e, in ogni caso, la scoperta di tale comportamento sarebbe motivo di esclusione dalle comunità di sviluppo. Al contrario, nelle aziende commerciali raramente tale comportamento viene seriamente sanzionato e la dimostrata presenza di *Easter egg*⁽⁵²⁾ è stata giustificata come un'esigenza del programmatore di poter eseguire con più flessibilità le fasi di *debug* di un programma.

A sostegno, invece, di una più elevata sicurezza del software proprietario si osserva che⁽⁵³⁾:

- la disponibilità del codice sorgente accresce il rischio che un pirata informatico utilizzi una vulnerabilità sfuggita ai programmatori per attaccare i sistemi che ospitano quel particolare software;

⁴⁹ La crittografia moderna, a differenza della crittografia antica, afferma che la sicurezza del metodo dipende esclusivamente dalla segretezza della chiave e non dalla mancanza di conoscenza del particolare algoritmo di cifratura.

⁵⁰ All'interno del movimento Open Source vi sono diversi gruppi di volontari che hanno creato "mailing list" e "forum di discussione" dedicati agli aspetti di sicurezza dei sistemi e delle applicazioni. Il loro lavoro è principalmente rivolto ai prodotti Open Source per la evidente facilità di analisi anche se, mediante l'utilizzo di "testing packages" e pratiche di *reverse engineering*, vengono talvolta controllati anche i sistemi proprietari.

⁵¹ Con il termine *trapdoor* (o *backdoor*) ci si riferisce al codice inserito all'interno di un programma con funzionalità "nascoste", note solo al suo ideatore, in grado di fornire, preservando le funzionalità originali, un accesso illegittimo a un sistema informatico.

⁵² Con il termine *Easter egg* si intende codice non documentato nascosto nelle applicazioni il cui scopo è normalmente quello di garantire agli sviluppatori di poterne prendere il controllo digitando una sequenza specifica di tasti.

⁵³ In particolare le argomentazioni riportate sono state estratte da un intervento di Steve Lipner, manager del security response center di Microsoft, alla Conferenza RSA 2001 e riportato dalla stampa specializzata (<http://online.securityfocus.com/archive/12/176723>).

- la diffusione delle notizie sulle vulnerabilità espone i sistemi a un maggior numero di attacchi;
- nessuno ispeziona il codice Open Source alla ricerca di eventuali problemi di sicurezza perché tale attività è noiosa, richiede molto tempo e non rende celebri; di contro l'attenzione del fornitore ai problemi di sicurezza è continua, qualificata e remunerata;
- i sistemisti di solito preferiscono dedicare tempo a installare patch e a leggere files di log piuttosto che il codice sorgente alla ricerca di vulnerabilità;
- l'attività di *peer review* (con questo termine si intende l'attività di controllo del codice sorgente del software da parte di chiunque vuole occuparsene) funziona bene per gli algoritmi di crittografia, che sono semplici, piuttosto che sui sistemi operativi da 40 milioni di righe di codice;
- il modello di sviluppo dell'Open Source si presta con maggiore facilità all'inserimento, da parte dei programmatori malevoli, di *trapdoors* che possono rimanere nascoste a una revisione superficiale.

Le affermazioni sopra riportate si collocano all'interno di un dibattito di natura teorica. Di seguito si riportano alcuni dati empirici rilevati da fonti specializzate:

- 1) **le vulnerabilità** riscontrate a carico dei sistemi e delle applicazioni sono raccolte, pubblicate e documentate da diverse organizzazioni; analizzando i dati riportati nel database delle *Common Vulnerabilities and Exposures* (CVE) ⁽⁵⁴⁾, relativi all'anno 2001, si ricavano i seguenti dati:

Vulnerabilità documentate a carico di:

- *web server Apache*: 6
- *web server Microsoft IIS*: 13

- 2) **i virus** per ambienti DOS/Windows conosciuti all'inizio del 2001 erano circa 50.000. Tale numero comprende tutti i virus conosciuti comprese le loro varianti e quelli presenti esclusivamente in laboratorio sviluppati a scopo di studio. I virus effettivamente in circolazione, riportati mensilmente dal sito www.wildlist.org, sulla base delle segnalazioni di 70 esperti di sicurezza, sono mediamente 600.

Per gli ambienti Unix/Linux non sono conosciuti veri e propri virus ma una loro variante i c.d. *worms* ⁽⁵⁵⁾. Per gli ambienti Linux ne sono attualmente noti 5.

- 3) il sito www.securitystats.com riporta alcune statistiche **sugli attacchi ai siti web**. Nel periodo compreso tra l'agosto 1999 e inizio gennaio 2001, 8071 *web server* sono stati oggetto di attacco diretto da parte di pirati informatici (non si riportano dati riguardanti attacchi generalizzati quali quelli dei più recenti *worm* Code Red e NIMDA). Tali server ospitavano i seguenti sistemi operativi:

58.94%	Microsoft Windows NT e 2000
20.35%	Linux (totale delle varie distribuzioni)
8.25%	Sun Solaris

⁵⁴ Il *Common Vulnerabilities and Exposures* è una lista di informazioni relative alle vulnerabilità di sicurezza del software resa pubblicamente disponibile sul sito www.cve.mitre.org. Ogni informazione contenuta nel database CVE riporta i riferimenti alle fonti che hanno pubblicato la vulnerabilità (CERT, Microsoft, Bugtaq e altri); essa viene mantenuta aggiornata dal MITRE Corporation, organismo *not-for-profit*, finanziato dal governo federale statunitense, quale centro di ricerca e supporto sui temi dell'ICT per i maggiori dipartimenti federali. In particolare per la manutenzione del CVE è finanziato dal FedCIRC www.fedcirc.gov (The Federal Computer Incident Response Center).

⁵⁵ Un "worm" è un programma in grado di introdursi via rete su un sistema, compiendo azioni illecite e di replicarsi in modo autonomo su altri sistemi elaborativi raggiungibili dal primo. Non vengono catalogati come virus perché, a differenza di questi ultimi, non hanno bisogno di un agente "portatore".

- 4) J.S. Wurzler Underwriting Managers (www.jswum.com), una delle prime società a offrire assicurazioni contro i danni conseguenti ad atti di pirateria informatica, ha deciso di praticare un aumento dei premi tra il 5% e il 15% se i clienti utilizzano, per le loro attività su Internet, piattaforme Microsoft Windows NT invece che Unix, Mac OS X Server o Linux. La decisione della società si è basata sulle conclusioni di un'ampia attività di *assessment* condotta su un campione di aziende medio-piccole assicurate (⁵⁶).

Secondo taluni osservatori i dati su riportati possono essere influenzati sia da motivazioni collegate alla posizione monopolistica della Microsoft, sia dalla diversa risonanza degli eventi commisurata alla diversa diffusione dei prodotti Open Source e di quelli proprietari.

⁵⁶ Riferimento Gartner FirstTake FT-13-8624 dell'8.6.2001.

7 *Analisi dei costi connessi ai prodotti open source*

I sistemi elaborativi degli istituti bancari sono realtà articolate e complesse costantemente sottoposte alle sollecitazioni dell'innovazione tecnologica. In tale contesto, nel processo decisionale volto a indirizzare gli investimenti nel comparto ICT, i costi indiretti, i vantaggi economici nel medio-lungo periodo, la rispondenza agli obiettivi strategici aziendali possono assumere un rilievo anche maggiore rispetto ai costi diretti.

7.1 **Identificazione dei requisiti**

Al fine di operare una scelta che si dimostri economicamente conveniente lungo tutto l'arco di vita del prodotto adottato, è necessario procedere a un'attenta valutazione comparativa dei giudizi assegnati a ciascuno degli elementi qualitativi sotto riportati.

Stabilità e affidabilità del prodotto; tali qualità vengono, nel caso dei prodotti commerciali, misurate in base al numero di installazioni presenti presso realtà aziendali importanti, alle referenze specifiche del fornitore e ai casi di successo che si possono registrare. Nel caso dei prodotti Open Source, non formano oggetto di offerta commerciale, vanno valutati:

- **il grado di diffusione del prodotto.** Un'ampia diffusione equivale a una maggiore garanzia di persistenza della comunità di sviluppo, di affidabilità e di supporto del prodotto stesso; il prodotto sarà sicuramente più valido quanto più grande è il numero delle installazioni effettuate in aziende omologhe;
- **la robustezza della comunità di sviluppo e la sua vitalità.** E' importante assicurarsi che:
 - esista un processo ben definito che illustri come e da chi vengono prese le decisioni importanti nell'ambito del progetto;
 - vi sia un numero consistente di sviluppatori a garanzia di efficacia e tempestività nell'evoluzione del software e nella correzione dei malfunzionamenti (⁵⁷);
 - esista il supporto di "organizzazioni forti" che forniscano infrastrutture ed eventualmente risorse umane e finanziarie; l'interesse di grandi organizzazioni ovvero di società commerciali è chiara indicazione di buone prospettive per il prodotto.

Flessibilità e ciclo di vita atteso del prodotto; nel caso sia prevedibile la necessità di intervenire sullo specifico software, la disponibilità del codice sorgente garantita dai prodotti Open Source è un elemento di rilevante importanza a condizione che:

- i linguaggi adottati siano diffusamente conosciuti a garanzia di manutenibilità, accessibilità a un gran numero di sviluppatori e permanenza del prodotto sul mercato;
- siano chiare le linee evolutive a breve/medio termine;
- il supporto strutturato al ciclo di vita del software (controllo delle versioni, distribuzione delle attività, tracciabilità e gestione degli errori) sia conosciuto e ben utilizzato da tutti i partecipanti;
- esistano linee guida e standard per la produzione del codice e di una completa e chiara documentazione.

⁵⁷ Va tenuto presente che la valutazione quantitativa delle comunità di sviluppo è fortemente condizionata dal momento storico del prodotto; per un progetto appena avviato una grande partecipazione è chiaramente un'indicazione dell'interesse dimostrato dalla comunità, mentre per uno maggiormente datato potrebbe essere una indicazione del raggiungimento di una fase di stabilità a fronte della quale si devono garantire soltanto le correzioni e le implementazioni minimali.

Interoperabilità; il valore di tale parametro è più o meno elevato in funzione del ruolo del prodotto nello specifico ambiente elaborativo e dei piani aziendali a medio-lungo termine; è necessario comprendere quanto il prodotto in esame condizioni le componenti elaborative con le quali dovrà integrarsi; non vanno ignorati o sottovalutati i costi di adeguamento dei prodotti "complementari".

Aderenza agli standard; tale parametro concorre all'interoperabilità tra applicazioni e sistemi diversi anche a fronte di evoluzioni future dei sistemi.

Portabilità; è più o meno determinante in funzione della necessità/probabilità di dover utilizzare lo stesso prodotto su piattaforme diverse; è importante verificare che il prodotto sia stato reso disponibile dalla comunità di sviluppo in ambienti operativi diversi (Linux, Unix, Windows) o che sia documentato il *path* di "migrazione"; ancora una volta la chiarezza del codice, la documentazione e i linguaggi utilizzati possono garantire il *porting* dell'applicazione o prodotto Open Source su piattaforme diverse da quelle citate e meno diffuse (sono noti diversi adattamenti su sistemi operativi di nicchia, in particolare OS/2).

Sicurezza; è necessario assumere approfondite informazioni sulla rispondenza dei livelli di sicurezza garantiti dal prodotto al fine di minimizzare i rischi di esposizione del sistema informativo aziendale; sia per i prodotti proprietari che per quelli Open Source è irrinunciabile la consultazione dei siti specializzati che rappresentano una fonte d'informazione costantemente aggiornata sulle problematiche di sicurezza dei prodotti più diffusi; l'esame del codice dei programmi che trattano dati sensibili rappresenta un'attività costosa ma in alcuni casi indispensabile.

Gestibilità; misura di tale parametro sono la quantità e la qualità del software di gestione a corredo di un prodotto e la sua predisposizione a interagire con i più diffusi sistemi di System&Network Management.

Disponibilità di applicazioni; il valore da assegnare a tale parametro dipende dall'utilizzo dello specifico software che si sta valutando. Nel caso si tratti di una piattaforma di base, per esempio Linux, è necessario aver presente l'ambiente in cui verrà utilizzato (ci si riferisce alle considerazioni espresse nel capitolo 3). Nel caso si tratti di un ambiente di sviluppo, la grande disponibilità di codice Open Source scritto in Java offre indubbi vantaggi in termini di reperibilità di oggetti pronti all'uso.

Qualità del supporto e dei servizi; è un elemento che va ponderato in relazione alle politiche aziendali adottate per gli altri software, alla disponibilità a valutare le nuove opportunità offerte dalle modalità di supporto innovative tipiche delle comunità di sviluppo Open Source e all'offerta di mercato. Mentre è noto come orientarsi sulle modalità di supporto per quel che riguarda i prodotti commerciali, per i prodotti Open Source è necessaria una riflessione più approfondita.

Riprendendo le considerazioni espresse circa la possibilità di utilizzare uno stesso prodotto Open Source in modalità che vanno dall'utilizzo quale pacchetto chiuso sino alla possibilità di equipararlo in termini di interventi di modifica a uno sviluppo interno, diviene chiaro che la qualità del supporto e dei servizi va valutata anche alla luce del previsto utilizzo. In tutti i casi è possibile ricorrere sia a un approccio tradizionale, stipulando un contratto con un fornitore, sia uno più vicino alla filosofia Open Source, che quindi preveda la collaborazione con la comunità di sviluppo. Quale che sia la scelta, il fatto di aver deciso di adottare un prodotto Open Source fornisce la garanzia di disponibilità del sorgente che lascia comunque spazio a evoluzioni future con la possibilità di non dipendere, per il supporto, da alcun fornitore.

7.2 Il Total Cost of Ownership

Sono disponibili numerosi modelli di rilevazione dei costi complessivi di un sistema informatico attraverso l'elaborazione di un indicatore economico detto TCO (Total Cost of Ownership). In generale, tali modelli - che possono differire per ambiti di applicazione o modalità di analisi - fanno riferimento alle fasi principali del ciclo di vita del software.

Le principali voci di costo che compongono il TCO sono rappresentate:

- dai costi diretti, quali l'acquisto di hardware e software, la gestione, l'amministrazione e la formazione;
- dai costi indiretti dovuti alle attività improduttive degli utenti finali e ai tempi di fermo dei sistemi.

Sono stati di recente pubblicati diversi studi che mettono a confronto i costi di soluzioni proprietarie con soluzioni Open Source (⁵⁸). I risultati esposti tuttavia sono fortemente dipendenti da fattori interni all'azienda (presenza di competenze interne, ricorso all'outsourcing per alcune attività, etc.) e da fattori esterni correlati al particolare mercato nazionale (costi dei servizi professionali di consulenza).

Gli studi offrono, pertanto, solo un punto di partenza per la comprensione dei reali costi ICT. Ogni azienda dovrebbe creare un proprio modello dei costi che sia rappresentativo dello specifico ambiente operativo interno e dei vincoli esterni indotti dal mercato.

Il modello proposto nelle pagine che seguono è stato elaborato quale strumento di paragone per una banca che intenda confrontare, per una specifica esigenza, soluzioni proprietarie rispetto a soluzioni basate su software Open Source ed è stato elaborato in modo da raggiungere una granularità delle informazioni tale da evidenziare i principali elementi di raffronto emersi nell'ambito dei lavori del gruppo. Per la sua definizione è stato, in particolare, esaminato quanto riportato nel documento "Free Software/Open Source: Information Society Opportunities for Europe?" redatto dal "Working Group on Libre Software" su iniziativa dell'Information Society Directorate General of the European Commission. Per ciascuna delle fasi individuate vengono riportate le principali voci di costo da valorizzare in base agli specifici scenari che debbono essere confrontati. Tenuto presente che le esigenze aziendali in termini di applicazioni software non sono completamente soddisfatte dai soli prodotti di mercato - siano essi proprietari o Open Source - ma si completano con un consistente patrimonio di software sviluppato internamente (eventualmente da personale esterno), lo schema proposto è comprensivo degli elementi utili al confronto tra:

- le due possibili modalità di acquisizione di un prodotto informatico: acquisto di un pacchetto commerciale vs. l'adozione di un prodotto Open Source;
- i costi di uno sviluppo interno rispetto a quelli di uno sviluppo condotto in modalità Open Source collaborando con una comunità aperta a contributi esterni.

In questo secondo scenario la valutazione comparativa dei costi non è esaustiva di tutte le problematiche legate alle modalità di sviluppo Open Source; esse, infatti, risultano decisamente alternative rispetto alle tradizionali regole della *software engineering* e quindi sono da valutare principalmente con riguardo agli aspetti organizzativi. Si è ritenuto, comunque, opportuno fare cenno a tale scenario per il crescente interesse che le modalità di sviluppo Open Source stanno suscitando presso le società di produzione del software e più in generale tra gli studiosi della materia, che vedono, nella progressiva capacità di interazione tra gruppi sociali diversi, nella diffusione di strumenti per il lavoro collaborativo e nella cultura della condivisione dell'informazione, la possibilità di affermazione di nuove forme di organizzazione del lavoro che, mutuando la struttura delle comunità Open Source, siano in grado di esprimere gli elevati livelli di produttività tipici di tali comunità.

⁵⁸ IDC White Paper sponsored by Microsoft "Windows 2000 Versus Linux in Enterprise Computing" (www.microsoft.com/windows2000/docs/TCO.pdf) Cybersource "Linux vs. Windows - Total Cost of Ownership Comparison" (www.cyber.com.au/cyber/about/linux_vs_windows_tco_comparison.pdf).

Fasi	Principali voci di costo
Raccolta e valutazione delle informazioni	
Valutazione e definizione dei bisogni aziendali e delle possibili soluzioni tecniche disponibili.	Risorse interne/esterne impegnate nella ricerca, valutazione, verifica e certificazione dei prodotti.
Sviluppo della soluzione	
Vanno considerati due casi distinti: 1) Acquisizione di software pacchettizzato 2) Sviluppo di un'applicazione " <i>custom</i> "	Nel primo caso i costi sono relativi all'acquisto del pacchetto software, calcolato sul numero di installazioni, nel secondo caso al numero di risorse necessarie allo sviluppo dell'applicazione. In entrambi i casi vanno calcolati i costi per gli eventuali adeguamenti delle infrastrutture hardware e di rete nonché dei software complementari.
Messa in esercizio	
Attività di sviluppo di moduli d'integrazione con i preesistenti sistemi, di definizione ed esecuzione di <i>test plan</i> e di predisposizione di procedure personalizzate di installazione e distribuzione del software.	Risorse interne e/o esterne.
Training	
Formazione degli amministratori e degli utenti della procedura	Corsi di addestramento del personale.
Utilizzo	
Gestione ordinaria e supporto utenti	Preparazione del personale interno/esterno dedicato alla gestione di primo e secondo livello della soluzione.
Manutenzione	
Correzioni, aggiornamenti, evoluzioni del software e dismissione dello stesso.	Risorse interne/esterne Contratti di assistenza per la manutenzione correttiva Licenze di aggiornamento del software Attività di recupero e la migrazione dei dati fra un vecchio e un nuovo sistema.
Indisponibilità del sistema	
Fermo dei sistemi e attività improduttive degli utenti	Risorse interne/esterne per l'analisi e la correzione del guasto. Mancata operatività degli utenti.

Dalle caratteristiche dell'Open Source si possono trarre alcune indicazioni generali sulla distribuzione dei costi rispetto alle varie fasi del ciclo di vita del software.

Nel caso in cui il TCO sia calcolato mettendo a confronto i costi di adozione di un pacchetto software Open Source rispetto a quelli di acquisto di un prodotto proprietario, si rilevano a carico della soluzione Open Source:

- costi maggiori nelle fasi di relative alla:
 - raccolta e valutazione delle informazioni; infatti, a differenza dei prodotti proprietari per i quali si può contare sui fornitori per ottenere approfondimenti, indicazioni e referenze, per i software Open Source non è sempre facile raccogliere informazioni sull'esistenza di componenti potenzialmente utili alle proprie esigenze e laddove queste siano disponibili occorre, in ogni caso, condurre specifiche attività di verifica;
 - training; per il software proprietario sono spesso previsti corsi a catalogo mentre per quello Open Source tali corsi devono essere realizzati appositamente anche se inizia a emergere una offerta di corsi per taluni software a maggiore diffusione;
- costi minori nelle fasi di sviluppo e manutenzione della soluzione per i bassi costi iniziali di acquisto e le successive attività di manutenzione correttiva ed evolutiva, per la maggiore semplicità di adeguamento alle infrastrutture preesistenti e infine per il riconosciuto minore impegno di risorse hardware;

Per le fasi di messa in esercizio e di utilizzo, ove la componente maggioritaria dei costi è rappresentata dall'impegno di risorse umane, vi è una sostanziale equivalenza tra le due soluzioni.

Messi a confronto i costi di uno sviluppo proprietario rispetto ai costi di conduzione di uno sviluppo nell'ambito di una comunità Open Source, si rileva che in quest'ultimo caso vanno sostenuti:

- costi maggiori per le attività di analisi e pianificazione; la particolare modalità di sviluppo rende complesse tali fasi in quanto si deve operare nell'ambito di una comunità che deve condividere gli obiettivi di progetto e che rimane comunque non impegnata contrattualmente nel rispetto di tempi predefiniti di realizzazione; richiede, quindi, una maggiore e più intensa attività di monitoraggio, una costante attenzione allo svolgersi degli sviluppi e una presenza attiva nella comunità;
- costi minori per le attività di sviluppo, test, documentazione e manutenzione; ciò in quanto parte dello sviluppo è svolta da risorse esterne i cui costi non sono sostenuti dall'azienda. Peraltro la presenza di una molteplicità di soggetti direttamente interessati allo sviluppo di un medesimo software garantisce come è tipico di realizzazioni Open Source:
 - una maggiore adesione agli standard predefiniti;
 - la generazione di un codice "pulito", chiaro e documentato;
 - un maggior numero di soggetti coinvolti nei test e conseguentemente una maggiore qualità intrinseca del software prodotto.

Vi è una sostanziale equivalenza per le fasi di messa in esercizio e formazione degli utenti.

Per ciò che riguarda i costi indiretti dovuti ai tempi di fermo dei sistemi e alla mancata operatività degli utenti, le riflessioni sinora esposte e gli studi degli analisti riportano costi generalmente inferiori per le soluzioni Open Source in virtù della maggiore tempestività negli interventi correttivi resa possibile dalle modalità di sviluppo del software della specie.

8 *Le raccomandazioni e le best practice*

Le considerazioni contenute nei capitoli precedenti mettono in evidenza che l'Open Source è una realtà ormai presente nel mercato del software - sia per i prodotti disponibili pubblicamente su Internet, sia per le strategie commerciali di alcuni fornitori (⁵⁹) - in grado di offrire, in specifici ambienti operativi e con adeguati presidi organizzativi e tecnici, sensibili vantaggi. Va, pertanto, segnalato che l'Open Source può presentare opportunità per lo sviluppo dei sistemi informativi aziendali tenendo presente che è fondamentale:

- mantenere aspettative realistiche nei riguardi dell'Open Source che, come ogni tecnologia, è solo uno strumento attraverso il quale porre in essere strategie di *business*;
- adottare soluzioni Open Source solo dopo averne individuato precisamente l'ambito di utilizzo; previsti i costi diretti, indiretti e la compatibilità con gli investimenti a lungo termine; valutati i rischi e definiti i ritorni attesi;
- non trarre giudizi generali dall'esito di progetti sperimentali a basso livello di investimento;
- valorizzare la potenzialità competitiva nei confronti del software proprietario al fine di creare positive forme di concorrenza;
- prestare attenzione alla portata innovativa del fenomeno sul piano culturale, organizzativo e tecnico.

8.1 *Le best practice*

L'individuazione e l'applicazione di *best practice* può contribuire alla riduzione dei costi diretti e indiretti connessi a un nuovo prodotto o tecnologia. Con riferimento ai prodotti Open Source vi sono tre aree in cui l'adozione di specifiche *best practice* può aiutare a controllare tali costi: il processo di selezione, le risorse umane, le risorse tecniche.

8.1.1 *La raccolta delle informazioni*

La corretta applicazione delle metriche di selezione del prodotto presuppone competenze specifiche che siano in grado di interpretare il complesso e variegato mondo dell'Open Source – sfavorito, tra l'altro, dalla mancanza di agenti di diffusione - e consentano, pertanto, di effettuare un obiettivo confronto con i corrispondenti prodotti proprietari.

La raccolta delle informazioni costituisce, di fatto, la prima vera differenza nel processo di scelta di un prodotto. Nel capitolo precedente, nella prima delle fasi del ciclo di vita del software era riportato, infatti, un maggior onere, a carico dell'Open Source, per la ricerca e valutazione delle informazioni.

E' fuori dubbio che, a fronte di un maggior sforzo iniziale e in virtù della disponibilità del codice sorgente e della documentazione, è possibile ottenere informazioni più complete e veritiere, perché non condizionate da logiche di tipo commerciale, e che al maggior onere iniziale corrisponde una miglior conoscenza del prodotto che può facilitare le fasi successive di gestione e supporto. Ma non va sottovalutato tale impegno, che in determinate circostanze, può, di fatto rappresentare un fattore di inibizione nei confronti dei prodotti Open Source.

⁵⁹ Già oggi molte aziende si trovano, o si troveranno presto, in situazione di "utilizzo trasparente" di software Open Source - Ref. Nikos Drakos, 02C, SPR3, 4/02 (Can Open-Source Software Save You Money?) "By 2003, 90 percent of all IS organizations will knowingly or unknowingly leverage open-source software within their software solution sets (0.8 probability)".

L'esigenza di trovare modi per garantire all'Open Source di competere sullo stesso piano e a parità di condizioni con i prodotti proprietari è diffusamente percepita da tutti coloro che a vario titolo si sono occupati di Open Source (⁶⁰). La proposta più ricorrente è quella di creare un catalogo dei prodotti Open Source nel quale vengano riportate le informazioni principali, le valutazioni, gli eventuali risultati di test effettuati, esiti di *benchmark* e casi di adozione. Uno dei primi esempi è il "bouquet du libre" curato in Francia dall'ATICA (Agency for Information and Communication Technology in Administration).

Il valore di tale catalogo è tanto più ampio quanto più indirizza gli interessi specifici di un settore. A tal riguardo la costituzione, a cura di associazioni di categoria, di un osservatorio tecnologico, che raccolga e metta a fattor comune tutte le informazioni rilevanti avrebbe il vantaggio di creare, in tempi brevi, una vasta e diffusa conoscenza della materia ed eviterebbe la dispersione delle esperienze e la duplicazione delle attività tecniche di verifica.

Il costo di una tale attività verrebbe ripagato in termini di conoscenza, di maggiori alternative tra le quali orientare le proprie scelte tecniche e di aumento della capacità contrattuale.

8.1.2 Le risorse umane

Una riflessione particolare meritano gli aspetti concernenti le risorse umane che, come evidenziato in diversi modelli di TCO, rappresentano una componente rilevante del costo complessivo di un sistema informatico.

L'introduzione del software Open Source in un'organizzazione non è un evento privo di conseguenze. Richiede, pertanto, un significativo cambiamento dei comportamenti perché possa essere integrato e accolto nell'organizzazione stessa.

In generale, l'introduzione di un nuovo software in una realtà complessa pone il problema della verifica dell'integrazione e della compatibilità con l'esistente. Spesso questa funzione è svolta da presidi centralizzati che governano l'architettura del sistema informativo e ne garantiscono una coerente evoluzione. Questo tipo di attività (in modo analogo a quanto avviene per la ricerca di un prodotto o di una soluzione), nei contesti tradizionali è spesso svolta in collaborazione con il fornitore, che mette a disposizione le competenze e il supporto per arrivare a definire le corrette modalità di inserimento e integrazione. Va comunque considerato che è prassi comune nelle aziende creare competenza interna, anche a fronte della presenza di un fornitore, per le successive attività di gestione.

Nel caso dell'Open Source, ancora una volta viene a mancare l'interlocutore di riferimento (si escludono da questa trattazione quei prodotti entrati a far parte dell'offerta commerciale di alcuni fornitori), ma lo sforzo di maggior approfondimento delle caratteristiche attuato in fase di reperimento della soluzione, ed eventualmente dell'analisi di esperienze note e documentate, viene in questo caso in aiuto perché molte delle conoscenze necessarie sono già state acquisite.

Non è comunque trascurabile il cambiamento nella modalità di interazione con il "fornitore" che evolve verso un rapporto paritetico, basato sulla consuetudine di lavoro con la comunità, sulla piena adozione dei principi di collaborazione e scambio di informazioni ed esperienze. Ciò inevitabilmente richiede un cambiamento importante: è necessario predisporre a una impegnativa collaborazione con entità esterne, le comunità di sviluppo, per le attività di acquisizione del software e delle informazioni e di restituzione delle eventuali modifiche correttive apportate; il software Open Source va considerato infatti un "bene collettivo", che rischia di essere impoverito

⁶⁰ Il documento del PITAC (comitato di esperti incaricato di fornire al Presidente degli USA e alle Agenzie federali interessate gli indirizzi strategici su tutte le aree dell'ICT) individuava tra gli ostacoli da superare, al fine di porre l'Open Source in grado di competere a un livello paritetico con le soluzioni proprietarie, la mancanza di un inventario dei progetti della specie.

da comportamenti di puro utilizzo (*free riding*).

Per altro verso la familiarizzazione con la cultura dell'Open Source fondata sui principi della collaborazione tipica di Internet, sullo scambio di esperienze, sul confronto culturale e sulla capacità di comunicazione può rappresentare un fattore di crescita professionale delle risorse aziendali e un arricchimento del sistema nel suo complesso.

8.1.3 Le risorse tecniche

Il software costituisce ormai uno dei più rilevanti *asset* aziendali. E' pertanto necessario che l'eventuale decisione di adottare software Open Source si ricolleggi a un quadro generale di regole che guidino il processo di selezione, adozione, utilizzo del software stesso e individuino i necessari presidi tecnico-organizzativi. In particolare, è necessario che vengano disciplinati:

- le modalità di acquisizione al fine di garantire l'integrità del codice;
- l'evoluzione in ambiente aziendale al fine evitare la proliferazione di versioni diverse dello stesso prodotto Open Source (devono essere predisposti opportuni ambienti di gestione del patrimonio software laddove non ve ne siano di preesistenti o questi non siano adeguati al software della specie);
- il costante allineamento al lavoro delle comunità di riferimento;
- le politiche di sicurezza, sia per gli aspetti che riguardano le varie fasi del ciclo di vita del software (acquisizione del prodotto da enti certificati, controllo delle versioni, delle modifiche, degli ambiti di utilizzo), sia per gli aspetti tecnici. A tal proposito è bene ricordare che è disponibile una documentazione ricca di spunti, di informazioni, di operazioni tecniche già realizzate;
- le forme di partecipazione alle comunità di sviluppo (iscrizione alle *mailing list* delle comunità che seguono gli sviluppi dei prodotti adottati, aggiornamento della documentazione, raccolta di informazione sugli sviluppi in corso, etc.).

9 Il futuro dell'open source software

Per tratteggiare il possibile futuro dell'Open Source si ritiene utile riportare alcuni orientamenti rilevabili dalle iniziative dei maggiori protagonisti del settore informatico.

Gli elementi di maggior rilievo a favore dell'affermazione dell'Open Source sono:

- le **previsioni degli analisti** che, seppure con forti scostamenti nelle stime, indicano Linux in crescita sia nel mercato dei server che in quello dei client; ciò, unito al progressivo superamento delle resistenze culturali nei confronti dell'Open Source, probabilmente farà da traino agli altri prodotti della specie;
- l'**attenzione delle PP.AA.**, le cui iniziative potranno contribuire a creare la "massa critica" che garantisca all'Open Source una durevole presenza nel mercato (cfr. allegato A nel quale vengono riportate le informazioni più significative);
- i **progetti innovativi in corso** presso enti di ricerca e università che stanno realizzando, in modalità Open Source, le infrastrutture software per la realizzazione della c.d. "*the next-generation Internet*". Particolare rilevanza, in tale ambito, riveste il progetto GRID⁽⁶¹⁾. Alla realizzazione di questo software, già utilizzato negli Stati Uniti e nel Sud-Est asiatico in contesti scientifici e tecnologici si è, di recente, affiancato il progetto europeo DataGrid finanziato dall'Unione Europea con 10 milioni di euro, cui parteciperanno il CERN di Ginevra e altri partner come INFN, CNR, Datamat e ITC-IRST di Trento. Il middleware DataGrid, così come l'originario progetto Grid, è un complesso di standard e di software Open Source. Sebbene il progetto Grid sia nato con l'obiettivo di dare una risposta all'aumento esponenziale delle esigenze di calcolo e di condivisione dei dati espressa dalla comunità scientifica internazionale, l'impegno, anche finanziario, mostrato dai protagonisti dell'ICT (IBM, Sun, Microsoft, etc.) lascia supporre un prossimo utilizzo in ambito aziendale.

Non vanno, tuttavia, sottovalutate alcune tendenze che potrebbero, se confermate, rappresentare reali minacce alla sopravvivenza del modello di sviluppo Open Source:

- la **brevettabilità del software**. In generale la pratica del brevetto del software può divenire un vincolo per chiunque sviluppi software; ma è inevitabile che l'Open Source sia particolarmente vulnerabile in quanto liberamente esaminabile e sviluppato da programmatori indipendenti e piccole aziende che non hanno il potere economico per proteggersi contro una causa per infrazione dei brevetti o richieste di pagamento di *royalties*. Attualmente la legislazione europea in materia non è del tutto chiara⁽⁶²⁾ e comunque difforme tra i diversi stati nazionali; nei tempi più recenti, in concomitanza con le attività della Commissione Europea per la preparazione di una nuova direttiva sulla brevettabilità del software, il dibattito sul tema si è fatto molto acceso. D'altronde si tratta di valutare attentamente le conseguenze che l'estensione della brevettazione a tutti i settori dell'economia contemporanea, compresi quelli immateriali, potrebbe comportare sul

⁶¹ Il Grid Computing è una tecnologia che ha avuto origine all'interno degli ambienti scientifici e accademici, dove da una decina d'anni viene usata per creare una 'griglia' (grid) di risorse elaborative eterogenee aggregate e condivise allo scopo di offrire supporto a progetti di grandi dimensioni e complessità. L'accesso alle enormi risorse di calcolo oggi distribuite nei vari ambienti scientifici, in modo semplice e trasparente, permetterà ai ricercatori di unirsi in laboratori virtuali internazionali, grazie ai quali, dal proprio computer si potrà usufruire di dati, potenza di calcolo, memoria e strumenti messi a disposizione attraverso la rete, senza bisogno di conoscerne la localizzazione.

⁶² Anche se la Convenzione Europea sui Brevetti esclude la brevettabilità del software e i brevetti software siano vietati da una specifica direttiva europea del 1991, l'Ufficio Europeo dei Brevetti (EPO) ha iniziato nel 1998 ad accettare domande di brevetto per programmi software e, alla data, ne ha riconosciuti più di 10.000.

futuro sviluppo economico, scientifico e culturale (⁶³);

- **modelli di *business* basati sulla chiusura di Linux** in versioni commerciali. Per ciò che riguarda Linux, in particolare, e quindi, in prospettiva, altri prodotti che guadagnino un forte interesse commerciale, potrebbero verificarsi tentativi di appropriazione con conseguenti sviluppi divergenti rispetto agli obiettivi originari delle comunità di sviluppo;
- **forme di protezione forte.** La Società Intel ha avviato un'iniziativa (Trusted Computing Platform Alliance) con l'obiettivo di realizzare una piattaforma informatica in grado di accrescere la sicurezza nei Personal Computer (⁶⁴). L'obiettivo è quello di rendere possibile l'esecuzione dei soli programmi certificati dal consorzio TCPA. E' naturale pensare che tali vincoli potrebbero frenare la diffusione dell'Open Source.

⁶³ In particolare vi sono pareri discordanti circa i benefici economici attesi dall'estensione di tale pratica. Lo studio di R. Hart, P. Holmes e J. Reid "The Economic Impact of Patentability of Computer Programs" Report to the European Commission, IP Institute http://europa.eu.int/comm/internal_market/en/intprop/study.pdf conclude che "*The theoretical and other economic literature does not demonstrate, indeed casts doubt, on whether economic efficiency, i.e. overall welfare, is achieved by having or making computer program related inventions patentable*". I sostenitori della brevettazione insistono sull'importanza dei brevetti nella creazione della ricchezza e portano a sostegno delle loro tesi la ormai consolidata prassi degli uffici brevetti statunitense e giapponese, anche se vi è da segnalare che anche in questi paesi vi è un acceso dibattito circa la validità di tale pratica. Lo stesso Greenspan in un recente intervento (Il Sole-24 Ore del 12.6.2002) metteva in guardia circa la fragilità di un'economia basata sugli *intangible asset*.

⁶⁴ Correlata a tale iniziativa vi è lo sviluppo del progetto Palladium della Società Microsoft.

10 Allegato A - L'orientamento delle Pubbliche Amministrazioni

Le informazioni fino a oggi disponibili sulla diffusione di Open Source Software sono carenti e talvolta divergenti. Ciò è particolarmente vero con riferimento alla realtà della pubblica amministrazione in Europa, in quanto le analisi sulla diffusione dell'Information e Communication Technology nel settore pubblico europeo non hanno quasi mai riguardato l'impiego di software Open Source in tale ambito. Tra le lodevoli eccezioni si possono citare un report della Information Society Project Office della Commissione Europea (settembre 2000), in cui la Finlandia e la Francia vengono menzionati come esempi di Stati che perseguono l'obiettivo di un significativo incremento dell'uso di Open Source Software nel settore pubblico, nonché uno studio, promosso dal Governo francese (Novembre 2000), in cui si evidenzia come una delle linee strategiche più significative nello sviluppo dell'ICT in ambito europeo per il 2002 sarà l'incentivo all'uso di software Open Source nel settore pubblico ⁽⁶⁵⁾.

10.1 Indagine sull'Open Source Software e settore pubblico in Europa

Una prima rilevante ricognizione sulla diffusione del software Open Source nel settore pubblico europeo è stata realizzata dalla Commissione Europea, nel primo semestre del 2001, tramite un'iniziativa, denominata I.D.A. ("Interchange of Data Between Administrations"), che ha riguardato sei Stati: Francia, Germania, Spagna, Belgio, Italia e Svezia. Dall'analisi sono emersi alcuni interessanti spunti di riflessione.

Da un punto di vista generale, i paesi *leader* sono risultati Germania e Francia, la prima per le concrete realizzazioni portate a termine, la seconda per il forte impulso governativo a favore di soluzioni basate sull'Open Source. Sul piano quantitativo, dall'indagine è emerso che:

- l'uso di software Open Source è concentrato sui server, in molti casi grazie alla sostituzione di preesistenti installazioni Unix con Linux; a tal riguardo, la stima è che circa l'8% dei server è dotato di almeno una componente Open Source dominante (sistema operativo, *web server*, database);
- sulle postazioni di lavoro l'uso di software Open Source è particolarmente ridotto (circa l'1%).

Con riferimento a tale ultimo aspetto, è significativo rilevare che le motivazioni addotte per spiegare un utilizzo limitato non sono relative alla mancanza di funzionalità o alla scarsa qualità dei prodotti Open Source quanto alla necessità, da parte degli utenti finali, di apprendere l'uso di nuovi strumenti e alla difficoltà di integrare componenti realizzate con altri prodotti, tipicamente appartenenti all'ambiente Windows.

I dati quantitativi variano significativamente se si prende in considerazione anche il settore *education*; in tal caso, circa il 63% degli intervistati afferma di usare software Open Source, seppur limitatamente; i server con almeno una componente Open Source dominante salgono a circa il 30% e le postazioni di lavoro con sistema operativo Linux raggiungono la percentuale di circa il 3%. Inoltre, le stime per l'immediato futuro sono nel segno di un forte incremento.

Notevole interesse riveste l'analisi dei dati qualitativi dell'indagine. Al riguardo, i criteri che, a detta degli intervistati, sono risultati prevalenti per l'adozione di una soluzione Open Source sono nell'ordine: l'interoperabilità, la sicurezza, il rispetto degli standard e le funzionalità, mentre il costo occupa solamente il 6° posto, alle spalle anche della facilità d'uso. Tale risultato smentisce il diffuso convincimento che il principale fattore di successo di tale software derivi dalla sua gratuità.

⁶⁵ Il documento non riporta, peraltro, indicazioni circostanziate sulle iniziative in corso nei vari paesi.

Con riferimento ai prodotti più utilizzati, i risultati dell'indagine confermano gli attuali trend di mercato. Sul lato server, domina l'accoppiata Linux/Apache; in particolare, Apache ha uno *share* di circa il 60%, grazie anche dalle caratteristiche multi-piattaforma di tale prodotto. Più articolata la situazione sulle postazioni client. Ciò che va evidenziato è che, mentre sui server alcuni prodotti Open Source come Linux, Apache e Samba sono ormai "market standard", in ambito personal computer non si è ancora imposto un insieme di prodotti tali da coniugare facilità d'uso e funzionalità estese. Per questo motivo, se da un lato Linux come sistema operativo, KDE come interfaccia grafica e Mozilla-Netscape come *browser* sono prodotti ormai di largo consumo, *suite* come OpenOffice, pur se apprezzate in maniera crescente, scontano soprattutto la scarsa integrazione con i prodotti Microsoft, in particolare Ms-Excel.

10.2 La situazione attuale in Europa

10.2.1 Francia

La Francia è oggi uno dei paesi leader in Europa per l'adozione di software Open Source, grazie soprattutto al forte impulso governativo; è ormai patrimonio comune il concetto che i nuovi sviluppi debbano essere basati sulla trasparenza degli strumenti utilizzati e sull'uso di "open standard", non proprietari.

È interessante rilevare come la Francia si sia avviata in ritardo rispetto ad altri paesi europei sulla strada di Internet. Per favorirne la diffusione, fu lanciata nel 1998 un'iniziativa a livello governativo – denominata PAGSI (Programme d'Action Gouvernemental pour la Société de l'Information) - che aveva lo scopo di tracciare le linee guida per lo sviluppo e la diffusione di Internet, in accordo con quanto già avviato in ambito europeo. Sebbene tale iniziativa non facesse esplicita menzione dell'Open Source, era comunque inserita in un ambito più vasto di attività a livello governativo che comprendevano anche la disponibilità di risorse finanziarie per la ricerca e lo sviluppo di soluzioni software innovative, di cui l'Open Source rappresentava una delle priorità.

Gli sviluppi di tali iniziative sono stati rilevanti, al punto tale che in un rapporto ⁽⁶⁶⁾ del 2001 "per un'amministrazione orientata al cittadino" vengono menzionate come priorità l'uso di XML per lo scambio di dati e la preferenza per soluzioni software Open Source.

Ulteriori spinte in favore di tale software sono derivate dalle iniziative del Ministero dei Pubblici Servizi, il cui titolare ha dichiarato ⁽⁶⁷⁾ che:

- implementare soluzioni Open Source vuol dire aderire agli "open standard";
- nel caso di scelta tra soluzioni proprietarie e Open Source, comparabili tra di loro come costi, vanno preferite le seconde;
- la quota di risorse finanziarie destinate ai progetti Open Source saliranno dal 10% al 33% (circa € 1,1 milioni).

Un ruolo particolarmente significativo nello sviluppo dell'ICT nella pubblica amministrazione è svolto dal MTIC (Mission interministérielle pour le développement des Technologies de l'Information et de la Communication dans l'administration), un organismo interministeriale con finalità di supporto tecnico. Il principale compito di tale organismo, che dipende direttamente dal Primo Ministro, è quello di rinforzare il coordinamento tra i ministeri, fornire aiuto alle varie amministrazioni pubbliche nell'avvio di progetti di ICT e diffondere informative agli interessati tramite *newsletter* trasmesse via *e-mail* ai sottoscrittori (più di 4000). È sufficiente prendere visione delle principali sezioni presenti sulla *home page* del sito Internet di tale organismo per apprezzare la vastità dei temi trattati in relazione all'Open Source:

⁶⁶ Rapporto Carcenac, (reso disponibile il 19 Aprile 2001) condotto su impulso del Primo Ministro Jospin.

⁶⁷ Conferenza "Open Source in public sector day" tenuta il 5 marzo 2001.

- tecnologie e standard, in cui si sottolinea la stretta connessione esistente tra standard, Open Source e interoperabilità;
- meeting ed eventi, di cui molti dedicati all'Open Source;
- dossier di contenuto strategico e tecnico, tra cui una specifica sezione dedicata all'Open Source;
- “outils libre” ⁽⁶⁸⁾ che contiene una lista dei principali tools Open Source (circa 50 a Giugno 2001).

Un altro prestigioso organismo nazionale, attivo nello sviluppo e nella diffusione di software Open Source, è l'INRIA (Institut National de Recherche en Informatique et en Automatique). La missione di tale istituto è quella di favorire il collegamento tra i vari centri di ricerca francesi operanti nei settori dell'ITC. Tra le varie iniziative assunte, va segnalata la partecipazione alle attività di organismi internazionali di standardizzazione – come il W3C – e la distribuzione di software, gran parte del quale Open Source.

Ulteriori iniziative meritevoli di menzione in ambito pubblico sono:

- la migrazione a Linux di circa 400 server del Ministero della Cultura e della Comunicazione. È inoltre prevista una completa infrastruttura Open Source entro il 2005;
- la realizzazione con soluzioni Open Source del Casellario Giudiziario Nazionale, curato dal Ministero della Giustizia; al riguardo, verranno utilizzati Linux come sistema operativo, PHP come linguaggio per lo sviluppo di “pagine dinamiche” *web* e MySQL come database;
- l'adozione di software Open Source da parte di molte Università francesi, come la Louis Pasteur University, dove Apache è installato su più della metà dei *web* server, l'Università di Lione, in cui tutti i server e l'80% delle workstation sono equipaggiati con prodotti Open Source, l'Università di Artois, dove l'utilizzo di tale software riguarda il 95% dei *web* server e il 10% delle workstation, l'Accademia di Rouen e l'Istituto Nazionale delle Scienze Applicate in cui tutti i *web* server sono Open Source;
- lo sviluppo con prodotti Open Source di un'applicazione Intranet per la Direzione Generale delle Dogane, facente parte del Ministero dell'Economia, della Finanza e dell'Industria. Al riguardo, è rilevante notare che tra i più importanti requisiti di progetto vi erano la sicurezza e l'affidabilità e che il costo dell'investimento è stato stimato basso, compresi anche l'addestramento e l'amministrazione.

A completamento di questa sintetica rassegna delle principali iniziative intraprese a favore dello sviluppo del software Open Source in ambito pubblico, è importante sottolineare il forte coinvolgimento degli utilizzatori, il che rappresenta una condizione imprescindibile per il successo di tale software.

Al riguardo, la “French Free software users community”, che è rappresentata da varie associazioni e comprende anche esponenti di settori pubblici, svolge il ruolo di supervisore, all'occorrenza critico, nei confronti dell'azione governativa, con il compito di indicare i correttivi da apportare al fine di rendere più incisive le iniziative assunte. Rientrano in tale ambito i commenti relativi al rapporto Carcenac, già citato, a proposito del quale si è rilevato che sarebbe meglio proporre l'obbligo e non la raccomandazione a favore dell'adozione di “open standard”, e la spinta decisa a favore della cosiddetta “eguaglianza delle opportunità” tra soluzioni proprietarie e non. A tal proposito, l'AFUL (Association Francaise des Utilisateurs Libre) afferma che gli ITC manager pubblici sono liberi di scegliere soluzioni proprietarie, ad esempio nel campo dei prodotti di office automation, a patto che un'autorità terza, indipendente dai fornitori, certifichi l'aderenza dei documenti così realizzati agli standard non proprietari.

⁶⁸ In Francia Open Source Software è tradotto Logiciel Libre.

10.2.2 Germania

Anche la Germania appartiene al gruppo dei paesi leader in Europa per l'impulso dato allo sviluppo dell'ICT nel settore pubblico. Al riguardo, una delle iniziative più importanti è stato il lancio, nel 1999, del programma "Modern State – Modern Administration", avente l'obiettivo di promuovere la diffusione della tecnologia nella pubblica amministrazione e di offrire ai cittadini servizi sempre più informatizzati.

In tale ambito, una delle iniziative più significative è costituita dall'avvio di un organo di coordinamento denominato KBST (Koordinierungs und BeratungsStelle), facente capo al Governo Federale, con lo scopo di curare la diffusione dell' ICT nell'amministrazione federale.

Nel corso del 2000, tale organismo ha prodotto un rapporto in cui sono analizzate *best practice*, tipologie di sviluppo ed esperienze maturate nei vari settori dell'ICT; tale rapporto costituisce il ruolo di un modello di riferimento per la pianificazione di progetti di ICT. Tra le varie tematiche trattate, particolare rilievo è stato dato al software Open Source, di cui si sono sottolineati i vantaggi in termini di affidabilità e di sicurezza. In particolare, viene trattato il tema, molto attuale, della diffusione dei prodotti Open Source nell'area dell'*office automation*, finora sviluppatasi in misura ridotta rispetto ai server. Di fronte alla contestazione che più sovente viene mossa dagli utenti nei confronti di iniziative di tal genere, vale a dire i costi – espliciti e occulti – connessi alla necessità per gli utenti stessi di apprendere l'uso di nuovi strumenti, nel rapporto vengono analizzate le principali criticità connesse all'adozione di soluzioni proprietarie mono-fornitore, che è peraltro la situazione attuale nell'amministrazione federale:

- gli *upgrade* di software o le nuove *release* dei prodotti di office rendono l'hardware rapidamente obsoleto in termini di capacità di calcolo e di memoria di massa;
- nella grande maggioranza dei casi, i documenti sono salvati in formato proprietario e la compatibilità è garantita (quasi) sempre in avanti, (pressoché) mai all'indietro;
- l'uso di formati proprietari impedisce una migrazione "soft" verso un software competitivo;
- la richiesta del Governo tedesco di adottare prodotti "trasparenti in relazione al modo in cui operano" è in evidente contrasto con l'adozione di soluzioni proprietarie, di cui non è ovviamente noto il codice sorgente, il che solleva dubbi sulla effettiva affidabilità di tale software e sulla sicurezza intrinseca (vedi ad esempio le cosiddette *back door*).

A fronte di tali considerazioni, nel rapporto vengono analizzate le varie soluzioni Open Source, le problematiche tecniche connesse con la sua integrazione con soluzioni proprietarie e i costi derivanti da iniziative del genere. Le conclusioni evidenziano, tra l'altro, che:

- una strategia volta a consentire la progressiva uscita dalla dipendenza da uno specifico fornitore può sostanziarsi solamente nella coesistenza dei prodotti Open Source con altri prodotti commerciali;
- in una fase iniziale, ciò può determinare uno sforzo aggiuntivo per la necessità di supporto derivante dalla presenza di più prodotti;
- adottare un sistema operativo Open Source corredato di prodotti applicativi, sia Open Source che commerciali, può rappresentare una soluzione stabile, con un rapporto costi/benefici ragionevole, tale da garantire la salvaguardia delle risorse e caratterizzata da adeguati livelli di sicurezza.

Un altro importante contributo è stato fornito dal Dipartimento Federale per l'Economia e la Tecnologia, che ha prodotto nel Marzo 2001 la pubblicazione "Open Source Software, a guide for small and medium enterprises" articolata su due temi principali:

- la definizione di Open Source Software e dei correlati aspetti anche di natura legale (modello di sviluppo, principali progetti, tipologia di licenze);
- spiegazione pratica dei pro e dei contro, le caratteristiche “*user friendly*” di tali prodotti, i servizi e il supporto nonché un modello dei costi con due esempi: *web server* e Internet router.

A proposito di quest'ultimo punto, gli esempi riportati sul documento ipotizzano tre scenari per ciascuno dei due esempi indicati, vale a dire una soluzione completamente Open Source, una soluzione mista (sistema operativo proprietario e applicativi Open Source), una terza soluzione completamente proprietaria.

Su tale base, e tenendo conto dei soli costi di acquisto e di installazione, i costi minori sono risultati quelli della soluzione completamente Open Source, seguiti da quelli della soluzione mista.

A cura del "Research Institute for open communication system" è stato fondato un organismo denominato BerliOS che svolge una funzione di raccordo tra sviluppatori Open Source, distributori commerciali di servizi e settore pubblico. Per il raggiungimento di tali scopi, l'organismo mette a disposizione documentazione aggiornata sugli ultimi sviluppi Open Source, *news*, archivi di software e soluzioni mirate per particolari settori industriali o commerciali.

Significativa è anche l'attività parlamentare; in tale ambito, si segnala il dibattito sulla proposta, avanzata da distinti gruppi politici, sulla promozione dell'uso di software Open Source nell'amministrazione federale, con particolare riferimento al progetto "E-Government-BundOnline 2005" da completarsi appunto entro l'anno indicato.

Il Ministero delle Finanze ha recentemente realizzato un progetto pilota per la Intranet aziendale - che attualmente serve circa 15.000 utenti - basato su Linux, traendo spunto dalle seguenti motivazioni:

- la preesistente cultura aziendale UNIX;
- la possibilità di riutilizzare l'hardware già disponibile;
- la disponibilità di software (Linux e Apache) senza costi di licenza.

Ulteriori iniziative a favore dell'Open Source sono:

- la decisione assunta dal "German Federal Army" - uno dei maggiori centri di utilizzo di Open Source in Germania - di favorire ulteriormente l'adozione di piattaforme “open” e non proprietarie;
- il varo di un progetto pilota, da parte di alcuni settori della pubblica amministrazione, per studiare la fattibilità della migrazione dagli attuali sistemi Windows a Linux;
- la realizzazione di un sistema di Groupware per il BSI (Bundesamt für Sicherheit in der Informationstechnik) l'agenzia federale per la sicurezza dell'Information Technology; il sistema offrirà servizi di *e-mail*, gestione dei contatti e del calendario;
- la decisione assunta dal Bundestag di adottare, a partire da Gennaio 2003, esclusivamente sistemi operativi Linux.

10.2.3 Spagna

Fino a tempi recenti, non vi è stata in Spagna una decisa azione governativa a favore dell'Open Source, perciò le esperienze maturate sono state finora prevalentemente frutto di iniziative isolate. Ciò non ha, tuttavia, impedito un vasto successo di tale software, come testimonia il fatto che, secondo una rilevazione condotta dall'International Institute of Infonomics dell'Università di Maastricht ⁽⁶⁹⁾ (giugno 2002), la Spagna occupa il terzo posto nella graduatoria di

⁶⁹ http://www.infonomics.nl/FLOSS/report/FLOSSFinal_2b.pdf

sviluppatori Open Source in Europa (dopo la Francia e la Germania).

Ultimamente, l'atteggiamento della pubblica amministrazione verso tale software ha iniziato a evolvere nel segno di un crescente interesse, suscitato anche dall'avvio di progetti di ampio respiro volti a favorire, in generale, la diffusione dell'ICT. In tale ambito, rientrano due iniziative denominate "Strategic Government Initiative for the development of the Information Society" e "General plan for Administrative Simplification" che sono collegate a iniziative europee (progetto e-Europe), in favore della diffusione dell'ICT, in cui l'uso dell'Open Source viene proposto come una *best practice* a livello governativo.

Una rilevante iniziativa è quella avviata dal Ministero della pubblica amministrazione per la realizzazione di un portale unico per l'accesso a tutti i siti pubblici. La prima fase del progetto ha riguardato l'installazione di server Linux in affiancamento ai preesistenti server UNIX, mentre l'obiettivo finale è quello di utilizzare esclusivamente server Linux (circa 400 per 4000 utenti) e di estendere lo sviluppo di applicazioni Open Source. È infine interessante notare che, per quanto riguarda i client, si è deciso di mantenere l'ambiente Windows poiché gli utilizzatori hanno ormai familiarizzato con i prodotti MS Office.

Altre significative esperienze sono:

- l'utilizzo di Linux come sistema operativo nelle scuole della regione dell'Estremadura, su decisione del governo locale;
- l'accesso a Internet presso il Nuclear Security Council;
- il sistema per la distribuzione di documentazione e la consultazione "in diretta" delle assemblee al Senato;
- il servizio di *e-mail* e *web* presso il Ministero della Giustizia.

10.2.4 Italia

Anche nel nostro paese si registrano azioni volte alla diffusione di software Open Source nella pubblica amministrazione. L'AIPA (Autorità per l'Informatica nella pubblica amministrazione), ha fornito specifiche indicazioni a favore di tale software nel piano triennale 2000-2002.

Altre iniziative sembrano indicare un crescente interesse delle istituzioni pubbliche verso tale tema. Una prima iniziativa è stata assunta dal mondo accademico, come testimonia la "Open Source Convention" organizzata nel dicembre 2000 dalla Scuola Superiore Sant'Anna di Pisa; esperienze analoghe sono state portate avanti anche dalle regioni Trentino e Toscana.

Negli ultimi tempi, l'ulteriore crescita di interesse verso l'Open Source ha favorito l'avvio di varie iniziative a livello pubblico di rilevante spessore.

Nel piano triennale 2001-2003 dell'AIPA si è accentuata la presenza di progetti volti ad agevolare la diffusione del software Open Source e parallelamente l'utilizzo sempre più estensivo degli open standard, muovendo dalla considerazione che l'adozione di soluzioni alternative a quelle proprietarie poggia in larga misura sull'accettazione e sulla condivisione di standard internazionali. Fra le iniziative di maggior rilievo figurano:

- l'avvio di una convenzione con la Scuola Superiore Sant'Anna di Pisa per la realizzazione e la sperimentazione di un sistema di protocollo sviluppato in modalità Open Source in base alle norme tecniche emanate dall'AIPA stessa;
- il varo di tre gruppi di lavoro su: integrazione dei dati e interoperabilità dei sistemi tramite XML, firma digitale, accesso per i disabili ai servizi della pubblica amministrazione.

Il 26 febbraio 2002 è stato presentato al Senato della Repubblica il disegno di legge concernente le "norme in materia di pluralismo informatico e sulla adozione e diffusione del

software libero nella pubblica amministrazione".

Da ultimo, il Ministero per l'innovazione e le tecnologie ha dedicato un significativo spazio alle opportunità offerte dall'Open Source in occasione della pubblicazione delle "Linee guida del Governo per lo sviluppo della Società dell'informazione nella legislatura" (Giugno 2002) e ha recentemente istituito (Novembre 2002) una "Commissione per l'Open Source nella PA" che "effettuerà una approfondita analisi delle tendenze tecnologiche e di mercato e, confrontando le posizioni in materia dell'Unione Europea, dei maggiori Paesi industrializzati nonché dell'industria Ict (Information and communication technologies), fornirà a tutte le Amministrazioni Pubbliche, gli elementi di valutazione per le scelte e le strategie riguardo il software a codice sorgente aperto."

10.2.5 Svezia

La situazione in Svezia è caratterizzata da aspetti articolati: infatti, da un lato questo paese ospita (con la vicina Danimarca) il più vasto *user group* Linux del mondo, dall'altro l'uso di Open Source nella pubblica amministrazione è ancora piuttosto limitato.

Le iniziative più interessanti sono state finora realizzate in ambito universitario; la Umea University ha organizzato la prima *totally-internet* sessione elettorale per studenti (nel 2001) utilizzando un sistema equipaggiato con sistema operativo Linux, database MySQL, *web server* Apache e *server side scripting language* PHP.

10.2.6 Belgio

Fino ad oggi, in Belgio l'uso di software Open Source è stato in prevalenza dettato da specifiche esigenze e non spinto da una incisiva azione governativa. Peraltro, la situazione è in evoluzione, come testimoniano alcune iniziative avviate di recente, tra cui lo studio a favore dell'Open Source condotto dal "National Joint Committee on Barriers to the Information Society" che opera nell'ambito di una più generale politica di sostegno promossa dal governo federale a favore dell'ICT.

Particolarmente attiva in tale senso è la regione di Brussels, in particolare per merito dell'azione del CIRB (Centre d'Informatique pour la Region Bruxellois): in tale zona, lo sviluppo dell'ICT ha determinato un forte radicamento di cultura UNIX che sta svolgendo il ruolo di apripista per una progressiva diffusione di prodotti Open Source (Linux, SamBA, MySQL, Perl) ⁽⁷⁰⁾. Tra le iniziative meritevoli di attenzione si segnalano numerose conferenze ⁽⁷¹⁾ e altri incontri a fini divulgativi; la regione di Bruxelles include, inoltre, la conoscenza di Linux, Samba, MySQL, Perl nei bandi pubblici di gara, sebbene tale comportamento non risponda ancora a una "*stating written policy*".

Un'altra iniziativa rilevante è quella avviata dal National Belgian Army che, sin dal 1998, ha sviluppato varie applicazioni utilizzando componenti Open Source, come router Linux, emulatori di terminali Logics, *web server*, ftp server, database server, proxy Squid e virus scanning Amavis.

⁷⁰ È interessante peraltro notare che, anche in questo caso, gli ambienti di office automation sono in larga misura basati su piattaforma Windows e, al momento, l'indice di penetrazione di prodotti Open Source in tale ambito è piuttosto basso.

⁷¹ In occasione della conferenza "Internet and Society" (marzo 2001) un'intera giornata è stata dedicata al tema dell'uso degli "open standard" nella pubblica amministrazione.

10.2.7 Commissione europea

In occasione del simposio "Best practices in e-administration" tenutosi in Lussemburgo nel marzo 2001, il commissario europeo per l'Information Society, E. Liikanen, ha sottolineato l'importanza degli standard per l'accesso *on-line* ai servizi offerti dalle pubbliche amministrazioni. Muovendo da tale considerazione, lo stesso commissario ha annunciato l'avvio di uno studio sull'Open Source.

Recentemente, l'Informatics Directorate (DI) ha effettuato un'accurata ricognizione del suo sistema informativo per identificare le possibili modalità di evoluzione. Attualmente, le piattaforme adottate sono: Unix per i database server, Unix e Windows NT per gli application server, Windows NT per i document server e Windows con prodotti di Office per i personal computer.

In relazione alla possibile opzione Open Source, sono emerse alcune linee strategiche meritevoli di attenzione.

- l'uso di software Open Source sui personal computer per attività di office automation è considerato ancora prematuro, sebbene si convenga sulla necessità di seguire attentamente l'evoluzione dei prodotti Open Source anche in questa area. Al riguardo, occorre tener presente un problema aggiuntivo per il DI derivante dal fatto che molte applicazioni tuttora operative sono state sviluppate in modalità client-server per cui sarebbe necessario riscriverle in caso di adozione di sistema operativo Linux sui client;
- l'uso di Linux e SamBA su file e print server non presenta problemi di integrazione in relazione alla connettività con le workstations Windows ma può causare difficoltà nel reperire i driver per le varie periferiche utilizzate (stampanti, scanner, ecc.): per questo motivo non si consiglia un vasto uso di Linux su tali server;
- l'adozione di Open Source su *web* server, database server e application server sembra una strada più facile da percorrere (Apache e prodotti correlati). Al riguardo, occorre prestare attenzione all'integrazione tra strumenti di sviluppo e componenti server nel caso di migrazione verso soluzioni Open Source, tenuto conto che gli ambienti di sviluppo più diffusi (ColdFusion, Oracle, Java) sono per lo più caratterizzati da una forte integrazione tra componenti client e server.

Tra le altre iniziative a livello europeo si segnalano il passaggio da Unix a Linux per il sistema SYSTRAN del DI, la realizzazione del portale EIONET (European Environment Information and Observation Network) con il prodotto Zope e le pubblicazioni su tematiche inerenti l'Open Source curate dal DG Information Society Working Group e dalla Information Society Technologies.

10.3 La situazione in alcuni Paesi extraeuropei

10.3.1 Gli Stati Uniti

Gli Stati Uniti hanno una lunga tradizione nello sviluppo del software c.d. *public domain*⁽⁷²⁾ che ha di fatto creato già da anni una cultura di apertura verso il software prodotto al di fuori dei tradizionali modelli di sviluppo.

I primi utilizzi di Linux, al di fuori dei circuiti accademici, sono noti da diversi anni. L'U.S. Postal Office utilizza dal 1998 sistemi Linux per la lettura automatica degli indirizzi.

⁷² Il software definito di *public domain* non è protetto da alcun *copyright*. Siffatta modalità di rilascio del software, prevista dal sistema legislativo statunitense, riguarda gli sviluppi finanziati con fondi pubblici. Il software *public domain* è reso disponibile a tutti i cittadini americani senza alcuna restrizione e può essere oggetto di commercializzazione.

Una delle prime dichiarazioni pubbliche a favore dell'Open Source è stata presentata nel settembre 2000 dal PITAC (President's Information Technology Advisory Committee)⁽⁷³⁾.

Nel documento esposto al Presidente Clinton, l'Open Source veniva definito "*una strategia attuabile per produrre software di alta qualità attraverso la collaborazione di privati, enti pubblici e università. L'approccio Open Source consente al nuovo software di essere liberamente condiviso e consente agli utenti di modificarlo, studiarlo e arricchirlo di funzionalità*". Il documento raccomandava, tra l'altro, di includere nel budget federale del 2001 finanziamenti a favore di:

- ricerca e sviluppo di software Open Source per l'High-End Computing;
- elaborazione di processi di *procurement* governativo che favoriscano lo sviluppo dell'Open Source;
- analisi delle problematiche legali connesse ai modelli di licenza dell'Open Source al fine di raggiungere un'uniformità contrattuale per il software della specie.

Anche se non vi è un obbligo all'adozione del software Open Source in ambito governativo sono ormai più di duecento gli enti e le agenzie federali che hanno adottato soluzioni Open Source. Tra i più rilevanti troviamo la NASA che ha realizzato NAIS (NASA Acquisition Internet Service) un sistema per l'invio di *e-mail* di notifica agli utenti basato su Linux e MySQL, il National Weather Service che ha recentemente deciso di sostituire le workstation Unix sulle quali venivano elaborati i dati relativi alle previsioni del tempo con sistemi Linux (le motivazioni in questo caso non riguardano aspettative di migliori prestazioni ma derivano da valutazioni economiche che, a detta dei responsabili, riducono sensibilmente i costi di esercizio dei sistemi), l'U.S.Navy, il National Security Agency con lo sviluppo del già citato selinux, il Defense Department, la Federal Aviation Administration, il National Institute of Health.

10.3.2 L'America del Sud

Il dibattito intorno al tema delle soluzioni Open Source nella pubblica amministrazione sta coinvolgendo un numero crescente di Paesi.

Il **Brasile** ha già da tempo adottato soluzioni Open Source in alcuni settori della pubblica amministrazione (Ministero della Sanità e dell'Istruzione).

In **Venezuela** il governo ha recentemente annunciato che la preferenza sarà data a soluzioni basate su Open Source e che le soluzioni proprietarie verranno adottate solo nei casi strettamente necessari. Alla base della decisione vi è la crescente preoccupazione circa la spesa in licenze software, il 75% della quale va a nazioni straniere, il 20% ad agenzie di supporto straniere e solo il 5% agli sviluppatori nazionali. Il governo si impegna a combattere la pirateria del software e a eliminare qualsiasi applicazione priva di licenza dagli uffici governativi. La necessità di applicazioni dovrà essere soddisfatta da programmatori locali e tutte le applicazioni sviluppate per il governo saranno soggette a licenza GPL.

In **Perù** sono in discussione, a livello parlamentare, tre proposte che prevedono un forte impegno della pubblica amministrazione nell'adottare soluzioni Open Source e nel favorire la migrazione dell'attuale parco software verso soluzioni non proprietarie. Al riguardo, è interessante notare che una delle principali motivazioni addotte non è legata a considerazioni di natura economica bensì attinenti ai requisiti di integrità, riservatezza e accessibilità delle informazioni.

Analoghe proposte di legge sono in discussione in **Argentina**.

⁷³ Il PITAC è un comitato di esperti incaricato di fornire al presidente degli USA e alle agenzie federali interessate gli indirizzi strategici su tutte le aree dell'ICT.

10.3.3 L'Asia

L'Open Source Software viene considerato di fondamentale importanza da alcuni dei governi dei paesi in via di sviluppo in virtù del fatto che:

- l'accesso al codice sorgente incoraggia le capacità locali, promuove la conoscenza e l'innovazione stimolando nel contempo la locale industria del software; quest'ultimo obiettivo è particolarmente importante in quanto l'Asia contribuisce marginalmente, con circa il 5% (fonte IDC), al mercato mondiale del software;
- favorisce l'indipendenza dalle compagnie straniere e riduce, quindi, il flusso di denaro in uscita dal paese;
- allarga la base dei possibili utilizzatori in virtù di costi di acquisto accessibili a fronte di redditi pro capite generalmente inferiori a quelli occidentali;
- Linux e, in generale l'Open Source Software, potrebbe aiutare la nascita di una forte industria del software consentendo a tali paesi di recuperare rapidamente il forte ritardo nel settore.

Accanto a queste ragioni di carattere più generale i governi dei vari paesi sono motivati anche da ragioni di politica interna. In **Cina**, per esempio, il governo sta promuovendo lo sviluppo di Linux per limitare, da un lato la dipendenza da sistemi operativi chiusi, e dall'altro per poter esercitare un maggior controllo sulle infrastrutture tecnologiche nazionali. Il Ministero dell'Industria Informatica e l'Accademia delle Scienze sviluppano una versione nazionale di Linux, *Red Flag Linux*, derivata dalla versione Red Hat.

Il Governo di Taipei (**Taiwan**) ha dato corso al piano nazionale "National Open Source Plan", riconoscendo l'opportunità strategica dello sviluppo di software Open Source nel miglioramento della competitività dell'industria nazionale, oltretutto il raggiungimento dell'indipendenza dalle forniture di software di largo consumo. E' prevista l'istruzione di 120.000 persone nel campo del software Open Source.

In **Corea** diverse società operano su Linux scrivendo applicazioni e versioni locali dello stesso. La principale, HancomLinux, ha sottoscritto un contratto con il governo per la fornitura di 120.000 copie di HancomLinux.

In **Malaysia** nel novembre 2001 il governo ha deciso di utilizzare software Open Source in tutti gli uffici governativi.

In **Pakistan** il governo si è impegnato a ridurre la pirateria informatica, che in tutta l'area asiatica sfiora secondo la BSA in circa il 95% del software, Linux, e in generale l'Open Source Software, vengono considerati la chiave di volta di questa iniziativa.

Più in generale Linux si presta a essere utilizzato all'interno dei piccoli apparati elettronici dei quali l'Asia detiene un'alta quota di mercato. Nello scorso anno, più di 20 industrie giapponesi tra le quali Toshiba, Sony e NEC hanno raggiunto un accordo per lo sviluppo congiunto di un sistema operativo basato su Linux adattato agli apparati elettronici di uso domestico quali telefoni, lettori MP3 e Personal Digital Assistant (PDA).

11 Allegato B - Le strategie dei maggiori vendor

Vengono riportate le sintesi degli incontri tenuti con i fornitori sul tema dell'Open Source e delle loro prospettive in merito.

11.1 IBM

La società IBM ha da tempo assunto un deciso impegno nei confronti di Linux e in generale all'Open Source. E' del marzo 1999 il primo annuncio di tale strategia seguito, nel dicembre 2000, dall'impegno della società a investire un miliardo di dollari su Linux, investimento che nel gennaio 2001 la IBM annuncia di aver già recuperato. La richiesta da parte del mercato di soluzioni basate su questa piattaforma, dimostrata anche dal crescente numero di installazioni, è stata la ragione di tale scelta.

Le linee principali di tale strategia riguardano il sostegno alle comunità di sviluppo Open Source con risorse umane ed economiche; l'istituzione di 11 centri di supporto alle attività di migrazione delle applicazioni verso l'ambiente Linux messi a disposizione degli utenti e degli Independent Software Vendor; un sito (www.ibm.com/developer/linux) in cui sono trattati temi tecnici, pubblicati numerosi documenti, messi a disposizione strumenti e applicazioni liberamente prelevabili; l'alleanza con i maggiori distributori Linux (SuSe, Red Hat, etc.); i servizi di consulenza e assistenza forniti da IBM Global Services nonché i corsi di formazione.

Tutte le linee di sistemi IBM: zSeries (mainframe), xSeries (server Intel), pSeries (RS/6000), iSeries (AS/400) possono attualmente ospitare Linux e su tale piattaforma sono disponibili i più noti prodotti software della società (DB2, MQSeries, Tivoli, WebSphere, etc.).

Per ciò che riguarda le possibili aree di utilizzo di Linux, la società considera consolidati gli scenari relativi ai servizi infrastrutturali, all'area del supercomputing, agli ambienti di sviluppo, all'e-commerce, al web hosting e all'automazione delle agenzie periferiche. Nel prossimo futuro a tutto ciò si aggiungeranno prodotti e applicazioni enterprise (DBMS, ERP, CRM, SCM, etc.) che ne amplieranno la sfera di utilizzo. Notevole interesse si registra nell'utilizzo di cluster Linux, sia per l'high performance computing sia per l'alta affidabilità. Molto promettente è l'area del consolidamento dei server.

A parere della IBM i punti di forza di Linux sono:

- la robustezza; con riguardo al tema riporta i dati dello studio di John DeVale della Carnegie Mellon University "Measuring Operating System Robustness" nel quale, presi in esame 15 sistemi operativi della famiglia Unix, Linux emerge tra i più robusti;
- la sicurezza; a tal proposito valgono i risultati di un recente sondaggio dal quale risulta che il 75% degli intervistati ritiene Linux più sicuro di Windows;
- il TCO; esperienze già realizzate e documentate hanno dimostrato che il TCO di Linux è inferiore a quello di altre piattaforme.

Per ciò che riguarda i limiti generalmente riconosciuti all'Open Source la società ritiene che alcuni di questi siano in realtà ormai superati in quanto:

- il numero di applicazioni di ISV in ambiente Linux è già superiore a 2900 e cresce a ritmo sostenuto;
- i servizi e il supporto vengono ormai a far parte integrante dell'offerta di IBM Global Services;
- il modello economico dell'Open Source, in cui i vantaggi economici per le aziende non sono legati alla vendita delle licenze ma ai profitti derivanti dalla fornitura di

servizi, è vantaggioso sia per le aziende che per i clienti in quanto consente di ridurre i costi fissi lasciando più spazio agli investimenti in servizi e soluzioni.

11.2 Microsoft

La società Microsoft tra i punti di forza del software Open Source cita:

- l'estesa attività di controllo del codice sorgente del software da parte delle comunità;
- i bassi costi di acquisizione, anche se ritiene tale vantaggio poco influente considerato il fatto che il costo del software rappresenta, secondo le stime della società, solo il 5% del complessivo TCO dei prodotti; evidenzia che il software Open Source, essendo più facilmente modificabile del software proprietario, tende a far salire i costi di supporto e manutenzione; considera, infine, difficile creare economie di scala laddove non vi siano limitazioni alla modificabilità del software;
- la trasparenza del software; in proposito, la società, riconoscendo il crescente valore che gli utenti assegnano alla trasparenza, ha avviato l'iniziativa Shared Source con la quale consente a grandi clienti la consultazione del codice sorgente di Windows pur rimanendo non consentita la modifica e la redistribuzione dello stesso.

Tra i punti deboli menziona:

- i costi di manutenzione e supporto;
- il controllo degli sviluppi e la relativa responsabilità;
- i rischi connessi alla licenza GPL, in quanto contraria alla protezione della proprietà intellettuale (IP), in termini di:
 - probabili limitazioni a lungo termine al modello di *business* per il fatto che il software deve essere reso disponibile gratuitamente;
 - possibili impatti sull'IP dell'utente;
 - incertezza su chi è responsabile del prodotto e su chi dovrà rispondere nel caso di violazione dei diritti IP di altri;
- la possibilità di derivare un progetto da un altro (*forking*) in virtù dei termini della licenza GPL; evenienza questa che crea problemi di consistenza, predicibilità e compatibilità del software;
- i modelli di *business* di tipo *loss leader strategy* (la strategia commerciale che prevede la distribuzione in perdita di qualcosa di valore nella speranza che l'utente compri prodotti complementari dai quali ricavare profitti ad esempio l'hardware, i servizi o altro software proprietario).

Ritiene elementi ancora aperti al dibattito:

- la sicurezza, tema rispetto al quale la società ha riportato i dati di un'indagine condotta e pubblicata da Information Security Magazine che ha rilevato nei vincoli di budget e nella mancanza di competenze adeguate i maggiori ostacoli nella realizzazione di appropriati presidi di sicurezza. Ha, inoltre, esposto alcune analisi di Gartner Group secondo la quale:
 - non vi è correlazione tra il modello di sviluppo del software e la sicurezza;
 - sino al 2005 la sicurezza dei prodotti Open Source non supererà quella dei prodotti proprietari (probabilità 0,7);
 - il numero di partecipanti ai test può migliorare la qualità di tale attività, di contro, l'aumento del numero di partecipanti allo sviluppo si risolve spesso nella produzione di software meno sicuro;
 - la "Security Mobilization Initiative", di recente avviata dalla Microsoft, pone al centro della strategia aziendale i temi della sicurezza nel processo di pianificazione, sviluppo e integrazione dei prodotti;
- gli impatti economici sull'industria del software che è uno dei settori più dinamici dell'attuale economia globale;

- i possibili interventi governativi a favore dell'Open Source rispetto ai quali la Microsoft sarebbe particolarmente critica in quanto ritiene possano influire negativamente sulla competitività del mercato.

La società ha presentato un'interessante indagine condotta su 34 grandi aziende italiane dalla quale sono emerse alcune indicazioni circa l'utilizzo di Linux di seguito brevemente riassunte.

Le principali finalità d'uso:

<i>Web</i> server dedicati	43.2%
Server di sicurezza (proxy, firewall, VPN, autenticazione)	29.7%
Server e-mail	24.3%
Server di rete (DNS, caching, router, remote access)	24.3%
<i>Business</i> application server (CRM, ERP, SCM, etc.)	21.6%
Internet, Intranet, Extranet	10.8%

Le ragioni della scelta (erano ammesse risposte multiple):

Costo	64.9%
Affidabilità	35,1%
Sicurezza	35%
Scalabilità	27%
Altro	18.9%
Flessibilità delle licenze	10.8%

Il perché di Linux in azienda (erano ammesse risposte multiple):

Politica Open Source aziendale	35,1%
Disponibilità competenze interne	18.9%
Disponibilità sorgenti	18.9%
Consiglio del fornitore di servizi	13.5%
Facilità formazione delle competenze	10.8%

In conclusione la società ha dichiarato che la sua strategia nei confronti dell'Open Source è quella di mantenere un approccio bilanciato e responsabile, di difendere il valore della proprietà intellettuale e il modello commerciale di produzione e distribuzione del software.

11.3 Sun Microsystems

La società Sun Microsystems dichiara che la sua strategia industriale è da sempre stata impostata sui principi della cooperazione nella definizione degli standard e della competizione nella loro realizzazione.

Con riguardo al fenomeno dell'Open Source rileva che tra i punti di forza vi siano:

- l'accesso al codice sorgente e la modificabilità del software;
- la condivisione del codice e della forza lavoro che garantisce tempi rapidi per i test e la correzione del software con risultati positivi sulla qualità dello stesso;
- l'ampia partecipazione alle comunità Open Source degli utenti finali del software che riescono in tal modo a guidarne gli sviluppi verso le loro reali esigenze;
- l'importante ruolo nei processi di standardizzazione.

Tra i punti deboli ritiene vi siano:

- gli inesistenti rapporti personali tra i partecipanti alle comunità di sviluppo, fattore che potrebbe rendere caotico il processo di sviluppo;
- la diffusa convinzione che il valore del software sia strettamente correlato al suo costo;
- il numero troppo elevato di licenze Open Source;
- le competenze necessarie per la ricerca e raccolta delle informazioni.

La società, riconoscendo il valore della condivisione della conoscenza nel processo di creazione dell'innovazione, collabora attivamente con le comunità Open Source. In particolare ha consentito la nascita della comunità OpenOffice alla quale ha rilasciato il codice sorgente del prodotto StarOffice e garantisce il costante allineamento tra i due prodotti, partecipa attivamente alle attività di adattamento di Linux sulla sua piattaforma hardware proprietaria (UltraSPARC), è membro di OSDN (Open Systems Developer Network), del Free Standards Group e di altri importanti comitati impegnati negli sviluppi di Linux.

La Sun ha, inoltre, reso pubblici, protetti da un particolare tipo di licenza: la Sun Community Source License, i codici sorgenti di alcune sue tecnologie software (Solaris, Java e Jini). La SCSL accoglie alcune caratteristiche delle licenze Open Source pur mantenendo i diritti legati alla proprietà intellettuale. Infatti il codice può essere modificato ma è la Sun che detiene il diritto di decidere se tali modifiche possono essere rilasciate o meno, inoltre, non è possibile creare nuove versioni dei prodotti derivate dal codice Sun a causa delle clausole che coprono i test di compatibilità. La Sun dichiara di aver apposto tali limitazioni per garantire l'ordinato processo di innovazione mediante il coordinamento centralizzato, la chiarezza sulla responsabilità, il controllo sulla compatibilità.

12 Glossario

API

(Application Programming Interface). Funzioni software formalmente definite che consentono lo sviluppo di applicazioni consistenti con il programma che le fornisce.

Browser

Applicazione software basata su un'interfaccia grafica che consente di visualizzare le pagine HTML di *Internet* e del *WWW*.

C

Linguaggio di programmazione definito e realizzato da D. Ritchie nel 1972 nell'ambito degli sviluppi del sistema operativo Unix. E' oggi uno dei linguaggi di programmazione più utilizzati.

E-business

Electronic Business. Processo automatizzato di tutti i processi aziendali attraverso cui è possibile svolgere in forma elettronica, su rete, ogni tipo di attività aziendale (comprare, vendere, fornire assistenza on line, consultare listini e cataloghi in rete, ecc.). Il termine fu introdotto per la prima volta nel 1997 da IBM.

E-commerce

Electronic Commerce. Il "commercio elettronico", cioè la possibilità di acquistare prodotti e servizi on line, attraverso il *World Wide Web*. Il commercio elettronico è uno strumento strategico che permette alle aziende di ottenere maggiori ricavi con l'espansione del proprio mercato, con la fedeltà dei propri clienti, con la riduzione dei costi e una conseguente maggiore efficienza.

ERP

Enterprise Resource Planning. Il termine ERP è stato coniato all'inizio degli anni '90 e comprende applicazioni software che riguardano la gestione integrata di tutte le risorse che partecipano alla creazione dei prodotti/servizi di un'azienda. Ottimizzano la collocazione delle risorse aziendali e realizzano la fornitura di beni e servizi con la massima efficacia. I vantaggi che un sistema ERP può apportare sono numerosi, dalla qualità dei dati alla massima tempestività dell'analisi, alla trasparenza sulla gestione e sulla proprietà dei processi. Di solito i sistemi ERP vengono utilizzati e integrati con sistemi di database.

Extranet

Una rete simile ad Internet ma limitata nell'accesso a partner, fornitori o clienti di un'azienda, cui è stata fornita un'apposita password. Permette di condividere in modo semplice e conveniente informazioni e risorse.

Firewall

Il firewall (letteralmente "parete antincendio") è un sistema che impedisce gli accessi non autorizzati. In pratica è un sistema in grado di controllare l'accesso alle reti intercettando tutti i messaggi in entrata e in uscita. Il firewall, a seconda della configurazione e della tipologia, permette infatti il passaggio solamente di determinati tipi di dati, da determinati PC e da determinati utenti. Il firewall separa e protegge la rete interna, definendo e rafforzando le policy di rete. I computer esterni alla rete devono attenersi a una specifica procedura per ottenere l'accesso alle risorse, agli host e a tutte le altre informazioni. Se l'accesso viene autorizzato l'utente può passare, a patto che si attenga alla procedura definita dal firewall.

Freeware

Software concesso in uso gratuito e liberamente distribuito ma del quale, in genere, non viene fornito il codice sorgente.

HTML

Hyper Text Markup Language. Linguaggio utilizzato per preparare le pagine che devono essere

visualizzate dai *web browser*. Creato nel 1991, l'HTML è stato elaborato successivamente dal CERN di Ginevra. Esistono molte versioni, tutte comunque, devono passare l'approvazione del W3C.

HTTP

Hyper Text Transfer Protocol. È il protocollo di comunicazione di Internet che consente lo scambio di pagine scritte in HTML.

Intranet

Rete interna simile a Internet che consente l'accesso ai soli dipendenti dell'azienda o membri dell'organizzazione.

Mailing list

Un elenco di indirizzi di e-mail identificato da un singolo nome. Quando un messaggio di e-mail viene inviato all'indirizzo della mailing list, questo viene automaticamente ricevuto da tutti i componenti iscritti alla mailing list.

Router

Termine che indica un dispositivo che è in grado di leggere l'header del pacchetto di dati per determinare il percorso di trasmissione migliore. I router possono collegare segmenti di rete che utilizzano protocolli diversi.

Shareware

Software del quale è permessa la distribuzione ma, in caso di utilizzo, è richiesto il pagamento di una licenza d'uso.

TCO

Total Cost of Ownership. Modalità di valutazione degli investimenti nel settore informatico. Il TCO si calcola considerando i costi diretti e indiretti di un'azienda derivanti dall'acquisto di materiale tecnologico.

Unix

Sistema operativo sviluppato presso i Bell Labs agli inizi degli anni 1970. Disegnato per essere di dimensioni contenute e flessibile, è stato il primo sistema operativo a essere scritto in un linguaggio di alto livello (C). Grazie alle caratteristiche di portabilità, flessibilità e potenza è oggi uno dei più diffusi sistemi operativi in ambiente server.

Usergroup

E' un gruppo di discussione che comunica *on-line* attraverso Internet.

W3C

World Wide Web Consortium. Il W3C è stato creato per sviluppare l'utilizzo del *web* attraverso protocolli comuni che assicurino l'interoperabilità tra i diversi sistemi. È un consorzio industriale internazionale guidato dal MIT Laboratory for Computer Science (MIT LCS) negli USA, il National Institute for Research in Computer Science and Control (INRIA) in Francia e la Keio University in Giappone. I servizi offerti dal Consorzio includono: un archivio completo sul World Wide Web per sviluppatori ed utenti, implementazioni e codici di riferimento e diverse applicazioni per dimostrare l'utilizzo della nuova tecnologia. Oggi il W3C conta più di 410 organizzazioni.

XML

eXtensible Markup Language. È un meta-linguaggio standard per la modellazione di documenti e dati. XML è basato su SGML, ma con un insieme ridotto di caratteristiche che lo rende più idoneo alla distribuzione sul *web*.

Bibliografia

- Nikolai Bezroukov: *Open Source Software development as a special type of academic research*, (http://firstmonday.org/issues/issue4_10/bezroukov/index.html)
- A. Bonaccorsi, C. Rossi: *L'Economia degli Standard e la Diffusione delle Tecnologie. L'Open Source non è un Assurdo Economico*, Laboratorio di Economia e Management, Scuola di Studi Superiore Sant'Anna (<http://lem.sssup.it>).
- S. Evers: *An Introduction to Open Source Software Development* (<http://user.cs.tu-berlin.de/tron/opensource>)
- F. Grasso: *Il Software Open Source scenario e prospettive*, Autorità per l'informatica nella Pubblica Amministrazione (AIPA), Maggio 2002 (www.aipa.it/servizi%5B3/notizie%5B2/scenariooss.pdf).
- R. Hart, P. Holmes e J. Reid: *The Economic Impact of Patentability of Computer Programs*, Report to the European Commission, IP Institute (http://europa.eu.int/comm/internal_market/en/intprop/study.pdf).
- IDA (Interchange of Data between the Administrations) European Commission, DG Enterprise: *Study into the use of open source software in the public Sector*
- International Institute of Infonomics University of Maastricht, The Netherlands: *Free/Libre and Open Source Software: Survey and Study* (<http://www.infonomics.nl/FLOSS/index.htm>).
- C. M. Kelty: *Free Software/Free Science*, International Institute of Infonomics, Maastricht (http://firstmonday.org/issues/issue6_12/kelty/index.html)
- J. Lerner, J. Tirole: *Some Simple Economics of Open Source*, Journal of Industrial Economics, marzo 2001
- R. Meo: *Software Libero e Open Source*, Mondo Digitale, n.2, giugno 2002
- The MITRE Corporation: *A business case study of open source software*, (www.mitre.org)
- C. Shapiro, H.R. Varian: *Information Rules. A strategic Guide to the Network Economy*, Harvard Business School Press, 1999
- Working group on Libre Software: *Free Software/Open Source: Information Society Opportunities for Europe?* (<http://eu.conecta.it>)