

Il software libero: una questione etica

Simone Piunno

Indice

1	La rivoluzione del software	
1.1	Tecnologia della comunicazione e società	
1.2	La nuova tecnologia: il software . . .	
1.3	Nuovi analfabetismi	
2	I problemi posti dal software	
2.1	Un'economia diversa	
2.2	Prodotti e servizi	
2.3	Pirateria e diritto d'autore	
2.4	Effetto rete, interoperabilità	
2.5	Bundling	
2.6	La catena dell'informazione	
3	Software libero per la ricerca	
3.1	La due comunità	
3.2	Un campo di sperimentazione	
3.3	La stessa etica	
3.4	I finanziamenti pubblici	
4	Software libero per la didattica	
4.1	Insegnare informatica nel modo giusto	6
4.2	Il problema dei costi	6
4.3	Garantire l'indipendenza	7
4.4	Insegnare standard aperti	7
4.5	Spendibilità sul mercato del lavoro .	7
4.6	L'indagine nel "testo"	7
4.7	Il progetto del MIT	7
4.8	Documentazione libera?	7
5	Conclusione	7

Sommario

2 Questo documento è derivato da un intervento che è stato tenuto il 24 ottobre 2002 presso la Biblioteca Comunale Bassani di Ferrara in occasione di un incontro pubblico dal titolo *Il software libero come movimento culturale*.

3 Dopo una analisi dell'impatto culturale che il software sta avendo sulla nostra società ed una trattazione delle problematiche che questo fenomeno ci sta ponendo, vengono dati alcuni spunti di riflessione sul ruolo che il software libero ha in queste dinamiche.

5 Per finire viene approfondito il valore che il software libero sta avendo nel mondo della ricerca e quello che potrebbe avere prossimamente nella didattica.

1 La rivoluzione del software

1.1 Tecnologia della comunicazione e società

L'introduzione di ogni nuova tecnologia per lo scambio dell'informazione e del sapere ha sempre avuto un pesante impatto sulla vita quotidiana dell'uomo. In particolare è noto [10] che questo successe quando vennero introdotti:

- la scrittura, che mise fine alla tradizione dei cantastorie come mezzo per tramandare oralmente la memoria e il sapere, tolse il primato di senso principale all'udito, cambiò la nostra forma mentis e portò in primo piano il ruolo dell'autore;
- la stampa, che portò i libri nelle case della gente, rese possibile la nascita dei quotidiani e delle enciclopedie, tolse potere al clero e permise la nascita del concetto di copyright;
- il telefono, che rese possibile il grattacielo, la distribuzione in tempo reale delle notizie, il poter disturbare una persona in ogni momento facendogli trillare qualcosa sulla scrivania;
- radio e televisione, mezzi di propagazione di massa dal centro alla popolazione, furono la causa della standardizzazione culturale, ad esempio la diffusione della lingua italiana rispetto al dialetto.

Da questa piccola analisi possiamo dedurre quanto la tecnologia usata nella trasmissione dell'informazione possa impattare la cultura e la nostra vita quotidiana.

1.2 La nuova tecnologia: il software

Oggi stiamo assistendo all'esplosione velocissima di una nuova tecnologia che viene comunemente chiamata telematica. La telematica racchiude tantissime cose come ad esempio: internet, gli SMS, le enciclopedie multimediali, i videogiochi interattivi, i libri elettronici, la firma digitale.

Alla base di tutte queste cose diverse sta lo stesso strumento: il software. Sottovalutare l'importanza culturale, economica e politica del software è certamente un errore poichè sempre più esso pervade ogni istante ed ogni situazione della nostra vita.

C'è software anche: nella centralina di iniezione della nostra macchina, nel sistema di gestione automatica degli ascensori, sul nostro telefono cellulare,

nel sistema di controllo dell'aeroporto, nell'archivio delle cartelle cliniche all'ospedale, nella prossima carta d'identità formato smartcard, nei sistemi di transazione delle banche e delle carte di credito, nel televisore, nel sistema di votazione di camera e senato, in quello che prossimamente ci faranno usare per le elezioni, nel nascente sistema per la distribuzione dei libri, della musica, del video (ad es. e-books, acquisto online, etc.), ovunque.

1.3 Nuovi analfabetismi

L'effetto dirompente di ogni nuova tecnologia della comunicazione provoca sempre una divisione tra chi ha e chi non ha accesso a tale tecnologia. Questo è successo ad esempio con la scrittura ed il problema è stato risolto un secolo fa con l'introduzione della scuola dell'obbligo. Per telefono, radio e TV la divisione era semplicemente tra chi poteva e chi non poteva permettersi l'acquisto dello strumento, ma non c'era una reale difficoltà nell'impararne l'utilizzo. L'uso del software è però un'attività complessa che richiede un certo grado di istruzione specifica.

Quanti sanno usare un computer oggi? Ad un secolo di distanza dall'istruzione di massa ci troviamo di nuovo in una situazione di analfabetismo diffuso, anche se si tratta di un analfabetismo diverso. C'è chi sa leggere e scrivere ma non usare i computer, chi lo sa usare ma non programmare e chi ha una padronanza completa anche se non intuisce il senso profondo di tutto il fenomeno.

Il peso culturale dell'informatica assume proporzioni via via crescenti man mano che essa diventa strumento del comunicare, del lavorare, dell'imparare, occorre quindi insegnare l'informatica e mettere molta attenzione in come lo si fa poichè questo peserà in maniera rilevante sulla società delle prossime generazioni.

Purtroppo la percezione del problema è differente nei vari livelli di alfabetizzazione:

- per chi non sa usare il computer tutto questo sono chiacchiere incomprensibili e prive di significato: sembrano poco concrete e lontane nel futuro; magari si intuisce l'importanza dello strumento ma si ritiene che sia troppo tardi per impararlo e si finisce per dipendere dall'aiuto dei figli o dei nipoti (c'è una forte correlazione con la stratificazione generazionale).
- per chi sa usarlo ma non programmarlo quello che conta è il costo del software e la sua effi-

cienza. Eventuali comportamenti anomali del software, dovuti a difetti o intenzionale malizia dell'autore, vengono presi come dato di fatto, acriticamente, come se fossero la normalità.

- per chi sa programmare interviene una coscienza maggiore del problema, si capisce quando il software sta facendo qualcosa che non dovrebbe e se si tratta di un difetto incidentale o di una cosa voluta, ma spesso ci si ferma al livello tecnico della questione senza preoccuparsi troppo, perchè ci si sente in grado di dominare il mezzo.

Poche persone si occupano di approfondire il senso filosofico di tutto questo, purtroppo anche tra gli umanisti non è stato finora mostrato molto interesse.

2 I problemi posti dal software

2.1 Un'economia diversa

I parametri economici nella produzione del software sono completamente diversi da quelli tipici dell'industria [15]. Nel software non c'è materia prima e tutto ciò che serve per produrlo è il cervello di una nuova classe di lavoratori: i programmatori.

Si potrebbe erroneamente pensare che il software sia un pezzo di plastica, ad esempio un dischetto o un CD, ma questi sono soltanto supporti usati per trasportarlo: il software è una entità immateriale, composta di pura informazione, di pure idee, e può essere trasportata per via telematica senza spostare un solo atomo di materia (questa operazione viene comunemente chiamata download). Anche i libri sono software: un software speciale che viene normalmente trasportato su carta e che è fatto per il nostro cervello.

Una volta speso un capitale iniziale in ricerca e sviluppo, la produzione in serie del software si basa su una mera duplicazione del prototipo, a costi pressochè nulli (duplicare l'immateriale costa infinitamente meno rispetto a duplicare un manufatto). Nonostante questo, si tende a vendere il software come un bene normale, con un costo unitario che non ha nulla a che vedere con il costo reale di "una copia in più".

Siccome i costi sono tutti nella prima fase, prima dei proventi, i produttori di software di massa tendono a comprimere il più possibile gli investimenti iniziali, rilasciando software spesso ancora incompleti o bacati ed in seguito faticando a correggere

le imperfezioni che loro stessi hanno lasciato aperte: correggere costa lavoro intellettuale, enormemente più costoso della copia in serie. Come se non bastasse per il software non ci viene data alcuna garanzia, nè ci viene sostituito se dimostra difetti (ciò è legale poichè in realtà non è un prodotto ma una licenza d'uso).

La particolare struttura economica di questo mercato pone terreno fertile per la produzione del software libero, poichè il costo iniziale di sviluppo viene efficacemente diluito nel tempo e diviso tra molti partecipanti, che lo fanno perchè ne hanno bisogno, guadagnando dal non doversi sobbarcare tutte quelle spese e l'intero rischio d'investimento che dovrebbero sostenere ugualmente. Per quanto possibile lo sviluppo e l'uso del software libero vengono condotti contemporaneamente, quindi i costi di produzione sono coperti subito da chi lo usa proficuamente per le proprie attività.

2.2 Prodotti e servizi

In questo momento sta nascendo un nuovo modo di fare imprenditoria con il software, un modo che nasce e si sposa con il software libero. Se cerchiamo di dare un servizio che dura nel tempo invece di un scatola con dentro un CD, possiamo giustificare una spesa da parte del consumatore, una spesa che essendo ripetuta nel tempo può anche diventare maggiore.

Ciò che si acquista (la consulenza, l'assistenza, il supporto) non è duplicabile e quindi ha valore reale. Allo stesso tempo il flusso continuo di introiti si può utilizzare per finanziare lo sviluppo di altro software sul quale proporre di nuovo lo stesso tipo di contratti, etc.

Si tratta di un fenomeno molto interessante per i governi [4] perchè:

1. Nel campo dei servizi si genera maggiore occupazione, ad esempio le analisi finanziarie dicono che IBM e Microsoft hanno grosso modo lo stesso utile ma IBM occupa 10 volte più persone di Microsoft. La differenza è che mentre Microsoft produce solo software di massa, IBM invece fornisce anche hardware e servizi.
2. Le grandi aziende di software tradizionale sono praticamente tutte fuori dall'Unione Europea quindi i soldi spesi in licenze se ne vanno oltre oceano. Allo stesso tempo le competenze necessarie per creare questo tipo di imprenditoria rimangono confinate altrove, senza

generare crescita tecnologica locale. Viceversa le aziende di servizi sviluppano know-how e fanno circolare soldi che però rimangono sul territorio.

Nonostante sia un fenomeno ancora limitato ci sono società che stanno prosperando e crescendo velocemente all'interno di questa filosofia. Questa nuova concezione dell'economia informatica sta spostando il baricentro del mercato e questo è il motivo per cui il software libero ha anche un pesante impatto economico oltre che culturale.

2.3 Pirateria e diritto d'autore

La legge sul diritto d'autore, anche nota come copyright, fu introdotta nel XVIII secolo per tutelare la produzione intellettuale. Da allora i termini di tale protezione sono stati continuamente estesi ed ogni volta che Mickey Mouse sta per diventare di pubblico dominio, il congresso americano provvede [8] ad estendere ulteriormente il periodo.

Forse per l'incapacità di giustificare quel prezzo altissimo in confronto al costo di riproduzione, sia il software che la musica soffrono di endemica pirateria. Si stima [1] che il 50% del software installato in Italia sia installato illegalmente, senza che sia stata pagata la licenza.

Per proteggersi dalla pirateria, i grandi produttori di software insistono continuamente per inasprire [11] la legislazione in materia di copyright e così già ora in Italia abbiamo l'anomalia di una legge che indica come reato penale (con pene fino a 3 anni di carcere) quello che alla fine dei conti è un semplice reato contro il patrimonio.

Come se non bastasse in Italia quando compriamo un supporto vergine (es. un nastro o un CD) oppure uno strumento di riproduzione (es. un videoregistratore) paghiamo una tassa a priori [6], anche se useremo tutto ciò per duplicare materiale di nostra proprietà, come il filmino del compleanno.

I governi, non trovando alcuna spinta nella direzione opposta, continuano ad indulgere nei confronti delle richieste dei discografici e dei grandi produttori di software, senza però ottenere risultati reali poichè le percentuali di software piratato non accennano a scendere.

Tutto ciò ha certamente ricadute negative nei confronti di come i nostri giovani percepiscono la legge: la legge diviene quella cosa che cerca di impedirgli di fare qualcosa che percepiscono come eticamente lecito e forse a volte addirittura

eticamente giusto (per limitare lo strapotere delle multinazionali).

C'è chi crede che ci sia qualcosa di sbagliato [5] in questo perverso meccanismo, non nei suoi principi ma in come viene applicato. Il software libero rientra pienamente nelle leggi sul copyright, ma ne rappresenta una diversa interpretazione e utilizzo, dando una possibile via d'uscita.

2.4 Effetto rete, interoperabilità

È noto che il valore del software cresce al crescere del numero di persone che lo usano [15], un po' come il telefono è più utile (quindi ha più valore) quante più persone ce l'hanno, quante più ne posso chiamare o possono chiamarmi. Questo succede perchè anche il software è uno strumento di comunicazione come il telefono: io produco un documento in un certo formato e chi possiede lo stesso software potrà aprire questo documento ed utilizzarlo.

Se uso un software che utilizza formati segreti per i dati, potrò scambiare dati solo con chi usa lo stesso identico programma. Viceversa se il software che utilizzo scambia formati aperti (documentati, pubblicamente noti nei dettagli) allora ci potranno essere altri software compatibili. Ecco allora che diventa evidente il ruolo dell'interoperabilità [2].

L'interoperabilità tra software diversi è un bene prezioso per l'utente e per chi produce software che hanno quote di mercato piccole, mentre è deleteria per il produttore monopolista (poichè indebolisce il monopolio). I produttori monopolisti fanno tutto quanto è in loro potere per promuovere protocolli e formati di dati non interoperabili e possono anche giungere al punto di regalare licenze e giocare in perdita. Ad esempio Internet Explorer, Outlook Express e Media Player sono forniti gratuitamente ma contengono estensioni segrete che li rendono non interoperabili con gli altri prodotti analoghi e con le altre piattaforme. C'è anche chi sospetta che i monopolisti del software in alcuni casi abbiano incoraggiato e favorito la pirateria, per ottenere un uso di massa che in seguito avrebbe garantito la posizione dominante sul mercato.

Chi accetta questo gioco sponsorizza implicitamente il monopolio, ad esempio cosa succede se il modulo elettronico per la dichiarazione dei redditi viene distribuito dal ministero soltanto in un formato segreto per usare il quale si è obbligati ad acquistare un certo prodotto software? Oppure se, cosa che avviene veramente, il telegiornale della pubbli-

ca amministrazione è solo in un formato che ha gli stessi problemi?

Noi crediamo che l'interoperabilità sia un profondo valore culturale, e un preciso compito istituzionale per l'amministrazione pubblica. Usare formati non interoperabili e propinarli al prossimo è anti-etico perchè obbliga il prossimo ad usare e acquistare lo stesso software che usiamo noi, va contro il consumatore impedendo la concorrenza ed è discriminatorio perchè esclude una parte delle persone impedendogli di accedere, come una barriera architettonica.

L'interoperabilità ha anche dimostrato di essere un valore economico e tre esempi di successo dell'interoperabilità sono: internet, il web e lo standard GSM.

Il software libero garantisce l'interoperabilità poichè deve essere aperto e ispezionabile per definizione, inoltre esistono molti progetti di software libero che cercano per quanto possibile di raggiungere la compatibilità con le applicazioni commerciali.

2.5 Bundling

Un altro metodo che usano i grandi produttori di software per aumentare il valore del loro prodotto e rinforzare il monopolio consiste nel forzare l'inserimento del loro prodotto tra quelli preinstallati sul nuovo hardware che viene venduto. L'effetto per il consumatore è che il costo del computer sale anche se quel particolare software non lo abbiamo chiesto, è come pagare una tassa obbligatoria o se vogliamo è un meccanismo di coercizione.

Immaginate cosa succederebbe se l'attuale monopolista (chiamiamolo X-Soft per convenzione) fosse nel campo della televisione invece che in quello del software: ci sarebbero numerosi produttori di televisori (così come ci sono numerosi produttori di hardware) ma tutti i televisori sarebbero venduti equipaggiati con un telecomando X-Soft, che invece di avere tutti i pulsanti ha solo alcune cifre.

Per avere le altre cifre occorre acquistarle separatamente e saper smontare il telecomando per installarle (a volte i produttori di televisori non vi vendono le cifre mancanti nè l'assistenza necessaria, poichè hanno un contratto di esclusiva con X-Soft).

Come mai, nonostante il software libero rappresenti una alternativa efficace e di costo parecchio inferiore, i produttori di hardware non lo propongono preinstallato sui loro computer invece di accettare la pressione del monopolista? È una questione

aperta ma ci lascia immaginare quanto potere un monopolista del software possa avere.

2.6 La catena dell'informazione

Tutto ciò suona estremamente sinistro eppure questa è la situazione: si rischia che un solo attore possa dominare l'intera catena dell'informazione [2].

Microsoft già oggi possiede il 90% del software che è installato sui nostri PC, una delle reti di portali più visitati su internet (MSN), e sta andando a caccia di contenuti da veicolare (ad es. il codice di leonardo, enciclopedie, siti di notizie, etc.) e di telefonini su cui installare il proprio software. Microsoft ha anche iniziato a produrre hardware, per esempio la console per giochi Xbox, un modello di telefonino, libri digitali, tastiere, mouse, etc. Ma questo è niente, ci sono progetti inquietanti, ad es. Passport e Palladium, che ci piacerebbe approfondire ma escono dallo scopo di questo ragionamento.

Il software di Microsoft viene utilizzato dai governi degli stati, i quali diventano in qualche modo dipendenti da Microsoft. Viceversa il software libero è pluralista e trasparente per definizione e darebbe molte più garanzie.

3 Software libero per la ricerca

3.1 La due comunità

Molto software libero oggi è costruito da una comunità culturale di pari-grado, la quale è nata grazie ad Internet mentre quest'ultima si andava formando, principalmente nel mondo accademico [7]. A sua volta il software libero è stato utilizzato come piattaforma di riferimento per l'implementazione di tutte quelle infrastrutture che permettono ad internet di funzionare. Tutt'ora c'è software libero sul 90% dei server di posta elettronica, sul 95% dei DNS e su circa due terzi dei server web.

C'è quindi un forte intreccio tra quella comunità accademica che ha fondato e sviluppato Internet e tra la comunità di sviluppatori che oggi porta avanti il software libero: esse si autosostengono e si integrano.

3.2 Un campo di sperimentazione

Il software libero aiuta indubbiamente il progresso scientifico in campo informatico poichè è il campo

di verifica e studio delle teorie.

Come esempio di battaglie per determinare quale sia il metodo migliore per fare un certo pezzo di software possiamo citare la lunghissima diatriba pubblica in cui Andrea Arcangeli e Rick Van Riel si sono sfidati per determinare chi aveva ideato il miglior sistema per la gestione della memoria virtuale. I fatti, dopo circa 1 anno di sperimentazione sul campo reale, hanno dato ragione ad Arcangeli nonostante l'altro metodo sembrasse teoricamente più prestante.

Avere migliaia di utilizzatori con elevata competenza tecnica, pronti a sperimentare le proprie varianti ed a discuterne nei dettagli la validità è una opportunità molto utile per far nascere nuove idee e trovare i punti deboli delle soluzioni inventate.

Con il software tradizionale spesso non è possibile questa discussione aperta, poichè le aziende vogliono mantenere il segreto sul funzionamento interno delle loro applicazioni.

3.3 La stessa etica

In fin dei conti il software libero segue l'etica della teoria scientifica:

- secondo il principio di falsificabilità di Popper [14] anche se sono pochi quelli capaci di verificare la consistenza di una teoria è importante che per tutti sia possibile farlo, garantisce poi la comunità scientifica che si autocontrolla egregiamente. Allo stesso modo vale per il software libero: pochi sono quelli capaci di scrutinare il codice sorgente di un programma di terzi eppure questo basta perchè tutti gli altri si fidino di quello che è stato prodotto.
- il progresso consiste nella falsificazione di teorie precedenti: ogni nuova teoria scientifica nasce scovando i bug delle precedenti in particolari situazioni limite e stabilendo una nuova teoria che è una formulazione migliore (una patch o un lavoro derivato da codice pre-esistente).
- questo progresso nella conoscenza è possibile solo perchè si può continuare a ragionare partendo da quello che è già stato capito, teorizzato e reso pubblico dagli studiosi precedenti. Newton diceva che non avrebbe scoperto nulla se non avesse potuto appoggiarsi sulle spalle dei giganti.

Anche fuori dal mondo informatico i modelli di sviluppo del software libero sono adatti alla ricer-

ca: il bazaar e il rispetto dei pari come incentivo al lavoro sono tipiche trasposizioni [15] delle dinamiche che da decenni muovono la ricerca scientifica in tutti i campi. Chi fa ricerca trova il software libero assolutamente naturale all'interno dei propri schemi culturali.

3.4 I finanziamenti pubblici

La ricerca scientifica tende a produrre altro software (ad esempio per realizzare esperimenti o prototipi di applicazioni reali) e quando tale risultato è stato pagato con i fondi pubblici viene abbastanza naturale rilasciarlo pubblicamente con una licenza libera così che possa eventualmente essere riutilizzato dalla comunità nella sua interessezza.

4 Software libero per la didattica

4.1 Insegnare informatica nel modo giusto

Per via dei nuovi analfabetismi occorre mettere molta attenzione in come si insegna l'informatica: insegnare con strumenti di un solo produttore non è probabilmente una buona idea. È un po' come se per insegnare come si legge un quotidiano si facesse vedere solo Repubblica e si desse per scontato che sono tutti tabloid. Occorre invece insegnare su un modello neutro, o per lo meno su una molteplicità di modelli, così che lo studente possa acquisire l'elasticità necessaria per adattarsi ad ogni alternativa.

4.2 Il problema dei costi

Usare software non libero implica un problema di costi per lo stato ma ancora di più per gli studenti poichè il software che usa l'insegnante in classe diventa automaticamente il riferimento e gli studenti che vogliono approfondire o fare i compiti a casa sono forzati a comprarlo.

Gli studenti che non possono permettersi l'acquisto rimangono svantaggiati o finiscono facilmente nel copiare illegalmente il software, a volte addirittura incoraggiati dall'insegnante. Chiaramente tutto questo è profondamente diseducativo.

Come soluzione alternativa sono disponibili dei cosiddetti "democd" di linux che la scuola può dare gratuitamente agli studenti, i quali possono

usarli a casa loro senza modificare la configurazione del PC e senza installare alcunchè, per la gioia del papà.

4.3 Garantire l'indipendenza

È importante garantire l'indipendenza del corpo insegnante e dei programmi di studio [2]. Ad esempio, la patente europea del computer (ECDL [13]), che è riconosciuta e valida per l'assegnazione di crediti formativi, è composta di 7 moduli dei quali 6 insegnano praticamente solo prodotti Microsoft.

Inoltre, successivamente all'emendamento del senatore Cortiana, in cui nel luglio scorso si chiedeva di modificare la legge di riforma della scuola tenendo conto anche delle possibilità offerte dal software libero, Microsoft ha deciso di regalare un corso e due licenze a 5000 docenti italiani. Iniziativa nobile ma sarà un caso?

4.4 Insegnare standard aperti

Insegnare gli standard aperti aiuta gli studenti a dare il giusto valore alla compatibilità e all'interoperabilità e li rende culturalmente indipendenti da un produttore. Uno dei grandi problemi del presente è l'emarginazione informatica, chiamata anche *digital divide*, problema che si pone pesantemente per chi non può permettersi di acquistare un certo prodotto software (es. paesi sottosviluppati o famiglie poco abbienti) o per chi non può usarlo semplicemente perchè essendo disabile necessita di interfacce differenti (es. terminali morse). Insegnare standard aperti aiuta a combattere l'emarginazione.

4.5 Spendibilità sul mercato del lavoro

Oggi giorno sul mercato del lavoro informatico sono di moda le certificazioni, ovvero corsi eseguiti dalle grandi aziende con un esame che attesta le proprie capacità. La professionalità di questi corsi a volte è dubbia: se paghi molti soldi è difficile bocciarti e se non ci sono abbastanza promossi non ci saranno abbastanza evangelizzatori del prodotto che magari si sta promuovendo commercialmente.

Al contrario l'onore di avere il proprio nome tra gli autori di un software libero di successo e il poter mostrare il codice che si è scritto, è una dimostrazione di competenza notevole che si può proficuamente utilizzare in un colloquio di lavoro.

4.6 L'indagine nel "testo"

Quando si insegna programmazione, poter "mettere il naso dentro" programmi veri giocando a modificarli è la cosa più efficace che ci sia per imparare. Ci sono persone che grazie a questa possibilità hanno imparato a programmare da autodidatti, un po' come leggendo dei buoni libri si impara a scrivere meglio.

4.7 Il progetto del MIT

L'uso del software libero per insegnare cose diverse dall'informatica è ancora embrionale: a parte il lato economico e l'indipendenza da un singolo vendor non si sa se si possano ottenere vantaggi in termini di maggior efficienza o qualità del prodotto.

È invece probabilmente vero che la collaborazione e la condivisione del materiale tra istituti differenti possa aiutare a produrre testi migliori. In questo senso il MIT si è già mosso [9] con il progetto OpenCourseWare del quale prevede di effettuare il lancio ufficiale entro un anno e si pensa che tutti i corsi possano essere coperti entro il 2007. Questa iniziativa ha importanza rilevante anche per quegli istituti che si trovano in paesi sottosviluppati e che possono in questo modo tenere corsi con un livello di qualità altrimenti impossibile.

4.8 Documentazione libera?

C'è chi sta cercando di trasporre i concetti del software libero nella produzione della documentazione tecnica, utilizzando ad esempio le licenze FDL [3] e OpenContent [12]. Non si sa se in questo campo si possa instaurare un ecosistema che si sostiene economicamente e c'è molta discussione in corso, nonostante questo ci sono alcuni editori che propongono libri di questo tipo, ad esempio O'Reilly e Apogeo.

Esiste inoltre un progetto italiano di nome GNU-temberg che consiste nella raccolta e catalogazione di testi coperti da licenze libere, scaricabili da internet, e nell'organizzazione di centri stampa nelle principali città dove chi vuole può ottenere una copia cartacea a prezzo di costo.

5 Conclusione

In seguito a questa disamina dovrebbe risultare chiaro quanto sia importante l'effetto della tecnologia informatica sulla nostra società e in che modo

l'uso di software libero potrebbe essere una scelta etica importante.

Insieme all'associazione FerraraLUG abbiamo deciso di aiutare la Biblioteca Comunale Bassani di Ferrara a mettere a disposizione del pubblico una postazione Linux, perchè crediamo utile che ci sia uno strumento di emancipazione ed alfabetizzazione che permetta di prendere confidenza con questo mondo.

Speriamo che la collaborazione possa proseguire e che questo progetto culturale si dimostri interessante anche per gli umanisti. Prima o poi sarebbe bello se oltre agli ingegneri anche i filosofi, i giuristi e i politici si occupassero attivamente di queste tematiche.

Riferimenti bibliografici

- [1] BSA: *bilancio positivo nella lotta alla pirateria informatica, ma c'è ancora molto da fare.*, 2002.
- [2] R. D. COSMO, *Trappola nel cyberspazio*. 1998.
- [3] FREE SOFTWARE FOUNDATION, *Licenza per documentazione libera gnu*.
- [4] R. GHOSH, *Free/libre and open source software: Survey and study*, tech. rep., International Institute of Infonomics, 2002.
- [5] J. GILMORE, *Cosa che c'è che non va nella protezione della copia*. 2001.
- [6] *legge n. 93 del 5/2/1992*, 1992.
- [7] S. LEVY, *Hackers. Heroes of the computer revolution*, Doubleday, USA, 1984.
- [8] M. MANTELLINI, *Lessig, un eroe moderno*, InterLex, (2002).
- [9] MASSACHUSETTS INSTITUTE OF TECHNOLOGY, *Open courseware*.
- [10] M. MCLUHAN, *La galassia Gutenberg. Nascita dell'uomo tipografico.*, Armando, Roma, 1976.
- [11] D. MINOTTI, *Riforma del diritto d'autore: chi vince e chi perde*, InterLex, (2000).
- [12] *Opencontent license (opl)*.
- [13] A. I. PER L'INFORMATICA ED IL CALCOLO AUTOMATICO, *European computer driving license*.
- [14] K. POPPER, *Logica della scoperta scientifica*, Einaudi, Torino, 1970.
- [15] E. S. RAYMOND, *The Cathedral & the Bazaar*, O'Reilly, USA, 1998.