

e&m

LA RIVISTA
DI DIREZIONE AZIENDALE



SDA BOCCONI
Scuola di Direzione Aziendale

1

economia & management



AREA SISTEMI INFORMATIVI

DISTRETTI VIRTUALI: LO STRANO CASO DELLE COMUNITÀ OPEN SOURCE

di Andrea Ganzaroli

GLI ASPETTI LEGALI DEI CONTRATTI INFORMATICI

di Gabriele Faggioli

ESTRATTO DA ECONOMIA & MANAGEMENT N. 1 - GEN/FEB 2003

ETAS



sistemi informativi

DI ANDREA GANZAROLI

DISTRETTI VIRTUALI: LO STRANO CASO DELLE COMUNITÀ OPEN SOURCE

L'innovazione è raramente il prodotto del lavoro di un team di ricercatori chiusi all'interno di un laboratorio di Ricerca e Sviluppo, ma richiede la partecipazione e il coinvolgimento attivo di una rete di imprese.



DI GABRIELE FAGGIOLI

GLI ASPETTI LEGALI DEI CONTRATTI INFORMATICI

La definizione del rapporto cliente-fornitore è strumentale all'aumento della possibilità di integrazione delle competenze di tutti i soggetti coinvolti, con un indubbio vantaggio per l'azienda.

DISTRETTI VIRTUALI: LO STRANO CASO DELLE COMUNITÀ OPEN SOURCE

@ L'innovazione è raramente il prodotto del lavoro di un team di ricercatori chiusi all'interno di un laboratorio di Ricerca e Sviluppo, ma richiede la partecipazione e il coinvolgimento attivo di una rete di imprese.

DI ANDREA GANZAROLI
Docente Area Sistemi Informativi
della SDA Bocconi

L'innovazione nel campo delle tecnologie informatiche e della comunicazione ha avuto una forte caratterizzazione territoriale. La Silicon Valley ne è un esempio. La letteratura sui sistemi produttivi locali ci insegna che l'organizzazione su base spaziale di un settore non è spiegata dalla semplice ricerca di economie di scala e di scopo su base territoriale, ma per lo più da una riduzione dei costi di innovazione. Le recenti teorie sul Knowledge Management e sull'innovazione infatti ci spiegano che l'innovazione è raramente il prodotto del lavoro di un team di ricercatori chiusi all'interno di un laboratorio di Ricerca e Sviluppo, ma richiede la partecipazione e il coinvolgimento attivo di una rete di imprese – clienti e fornitori – e persone che condividono un comune progetto. Sempre queste teorie ci spiegano che l'innovazione non è il semplice risultato di uno scambio di conoscenze codificate o “scientifiche”, ma richiede anche la condivisione di conoscenze tacite o di contesto. Queste ultime, come noto, sono il risultato dell'accumulazione di esperienze all'interno di uno specifico contesto sociale, culturale e operativo.² Lo specifico combinarsi di queste due dimensioni della conoscenza

– tacite, che sono specifiche del luogo, e codificate – genera innovazione. L'organizzazione di un settore industriale su base territoriale è quindi spiegata dall'esigenza di ridurre i costi associati allo scambio delle conoscenze tacite o di contesto.

Le comunità *open source*, da questo punto di vista, rappresentano un'eccezione. Sono infatti comunità di sviluppatori software localizzati in diversi contesti territoriali e istituzionali che cooperano nello sviluppo di applicazioni attraverso l'utilizzo di media informatici e della comunicazione digitale. L'interesse verso queste comunità è spiegato dalla loro capacità di generare innovazione. Esse, infatti, hanno sviluppato prodotti innovativi quali Linux, che da molti viene considerato il principale competitor di Microsoft, e Apache, che è leader nel mercato dei web server.

Questo breve contributo si propone di esplorare i principali aspetti che caratterizzano il funzionamento di queste comunità, evidenziando gli elementi di contesto che facilitano una efficace cooperazione tra i loro membri. L'articolo si inquadra nell'ambito di una ricerca SDA³ il cui obiettivo è approfondire la conoscenza del fenomeno *open source* e definirne le potenzialità nel contesto italiano. Questa ricerca si avvale del contributo di tre Aree: Sistemi Informativi, Organizzazione e Pubblica Amministrazione.

Le origini delle comunità open source

Per comprendere il fenomeno *open source* è utile rivisitare alcuni fattori storici che ne hanno caratterizzato lo sviluppo. L'origine del modello ha avuto luogo attorno ai primi anni sessanta in quattro labo-

1. Per approfondimenti: I. Nonaka, H. Takeuchi, *The Knowledge Creating Company*, University Press, Oxford, 1995.
 I. Nonaka, N. Konno, "The concept of *ba*: building a foundation for knowledge creation", *California Management Review*, vol. 40, n. 3, Spring 1998.

2. Si veda al riguardo: E. von Hippel, "Sticky information and the locus of problem solving: implications for innovations", *Management Science*, n. 4, 1994.
 E. von Hippel, "Economics of product development by users: the impact of sticky local information", *Management Science*, 5, pp. 629-644, 1998; J. Nahapiet, S. Ghoshal, "Social capital, intellectual capital and the organizational advantage", *Academy of Management Review*, vol. 23, n. 2, pp. 242-266, 1998.

3. Chi fosse interessato ad approfondire l'argomento può contattare andrea.ganzaroli@sdabocconi.it. Sul tema dei software open source si veda Raymond, *The Cathedral and the Bazaar*, O'Reilly, 1999.

4. La fama del centro è legata a due curiose ragioni: l'aver prodotto molte delle innovazioni che caratterizzano i nostri moderni pc (vedi l'idea dell'interfaccia grafica) e il non essere stato in grado di sfruttarne il valore commerciale.

5. Per un approfondimento su questo tema si veda S.H. Lee, *Open Software Licensing*, www.opensource.mit.edu, 1999.

ratori di ricerca di AI (Artificial Intelligence), tre dei quali localizzati presso alcune delle principali università americane, e rispettivamente: MIT, Stanford e Carnegie Mellon. Il quarto, invece, era un laboratorio con scopi commerciali: il famoso centro di ricerca della Xerox a Palo Alto.⁴ Tutti, a partire dagli anni sessanta, hanno attratto le più brillanti menti nel campo dell'informatica a livello mondiale. Il clima di ricerca universitaria che caratterizzava questi laboratori favoriva la condivisione del prodotto del proprio intelletto. In altre parole, era buona norma tra i membri condividere il codice sorgente dei propri programmi. Lo scambio di codici sorgente non era fine a se stesso, ma percepito come occasione di apprendimento. La persona che riceveva in "dono" il codice sorgente, infatti, non si limitava a utilizzarlo per i propri fini, ma lo modificava e adattava alle proprie esigenze. Attraverso lo scambio, perciò, il codice sorgente si arricchiva continuamente di nuove funzionalità. Lo scambio, in altre parole, era il motore stesso dell'innovazione. Attraverso lo scambio i ricercatori dei laboratori apprendevano nuove conoscenze dalle esperienze e dalle competenze altrui e le integravano con le proprie.

L'istituzionalizzazione di una cultura

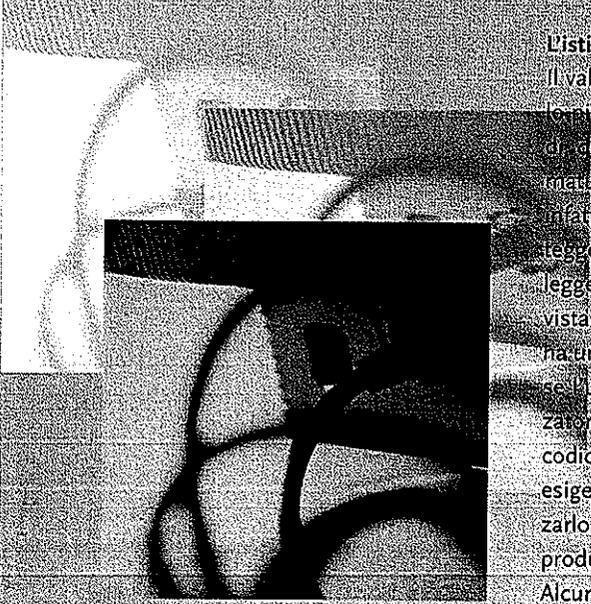
Il valore della condivisione, però, era per lo più locale. Era perseguito dai membri di questi laboratori di ricerca. Con la maturazione del settore del software, infatti, sempre più è invalso l'uso di proteggere il codice sorgente attraverso la legge sul diritto d'autore. Dal punto di vista dell'utilizzatore, questa operazione ha un impatto molto forte, specialmente se l'utente è molto competente. L'utilizzatore, infatti, non può più modificare il codice sorgente per adattarlo alle proprie esigenze specifiche. È obbligato a utilizzarlo secondo le funzionalità previste dal produttore.

Alcuni dei membri dei laboratori di cui

sopra, per favorire la diffusione di una cultura cooperativa nello sviluppo del software, fondarono la Free Software Foundation, il cui scopo era fornire un quadro istituzionale alla pratica del condividere i codici sorgente. La fondazione, infatti, ha istituzionalizzato la licenza *copy left*, che non è in antitesi con la legge sul *copy right*, ma ne è un'interpretazione "liberale". Secondo la licenza *open source*, infatti, un software è aperto/libero se risponde a due condizioni principali: è distribuito assieme al codice sorgente; l'autore autorizza la modifica del codice stesso. È importante notare che l'autore non rinuncia ai diritti. La paternità sulle righe di codice sviluppate è riconosciuta e difesa all'interno della comunità.

Esistono diverse versioni di licenze *open source*, che differiscono prevalentemente in termini di libero uso del codice sorgente. La più restrittiva è la GNU della Free Software Foundation. Se un codice sorgente è protetto dalla licenza GNU, ogni lavoro derivato deve essere anch'esso distribuito secondo le condizioni della licenza GNU.

Questo vale anche nel caso dell'integrazione tra codici privati e pubblici. La parte privata, in questo caso, diventa pubblica. La più liberale (Artistic License), invece, non pone alcun limite all'uso del codice sorgente. Esso, una volta modificato, può essere privatizzato e distribuito secondo le tradizionali licenze *copy right*. Esiste poi un certo numero di alternative intermedie. Per esempio, la licenza BSD, utilizzata dall'Apache Group, permette di privatizzare versioni modificate del codice sorgente a condizione che venga fatta esplicita referenza all'autore del pezzo originale. Questo modello di licenza, in altre parole, replica il modello di gestione dei diritti d'autore tipico dell'accademia, dove l'autore deve fare riferimento alle fonti che utilizza per sostenere le proprie argomentazioni.⁵



L'altro fattore abilitante della comunità: Internet

L'istituzionalizzazione della pratica del *copy left* non è sufficiente a giustificare l'evoluzione delle comunità *open source*. L'altro fattore che vi ha fortemente contribuito è stato lo sviluppo di Internet. La rete, infatti, ha consentito di estendere le comunità *open source* a livello globale reclutando adepti da tutto il mondo. Il primo a sperimentare pesantemente le potenzialità offerte dal nuovo mezzo nello sviluppo del software è stato Linus Torvalds, colui che, tra l'altro, ha sviluppato il kernel – il cuore operativo – del sistema operativo Linux. Il principale contributo di Torvalds, tuttavia, è stato l'invenzione di un modello di sviluppo in grado di sfruttare al massimo le potenzialità offerte da Internet quale nuovo ambiente di sviluppo cooperativo. Molto dello sviluppo *open source* prima di Linux era stato gestito in modo tradizionale: il software veniva progettato e sviluppato da una singola persona, o da un numero molto ristretto di persone. Una volta testate le funzionalità del software, questo veniva rilasciato alla comunità, che poteva utilizzarlo secondo le modalità permesse dalla licenza *open source*. L'interazione con l'utilizzatore finale era praticamente nulla. L'innovazione apportata da Torvalds è stata quella di cercare fin dall'inizio l'interazione con i potenziali utilizzatori del software attraverso la creazione di una larga comunità di "beta tester". Torvalds, nello sviluppo di Linux, si è preoccupato di fornire alla comunità non tanto un prodotto "perfetto", ma uno che funzionasse. L'incompletezza del software rilasciato e i possibili bug presenti hanno stimolato gli utilizzatori a fornire informazioni sui problemi riscontrati, possibili soluzioni e suggerimenti per migliorare le funzionalità e le performance del software. Il segreto di Linux è stato l'ascoltare i propri utilizzatori e rispondere in tempi rapidi alle loro esigenze e suggerimenti. Internet ha rap-

presentato lo strumento che ha consentito di massimizzare il numero degli utenti coinvolti nello sviluppo.

Conclusioni

Abbiamo iniziato il nostro articolo affermando che molto dello sviluppo nel campo delle tecnologie dell'informazione e della comunicazione ha avuto luogo in alcuni contesti territoriali specifici quali, per esempio, Silicon Valley, e che questo modello di sviluppo su base territoriale facilita la libera circolazione delle conoscenze tacite e quindi la capacità di innovare delle imprese localizzate in tali contesti. Le comunità *open source*, in questo senso, rappresentano un'eccezione: lo sviluppo delle tecnologie *open source* è basato, infatti, sull'interazione mediata dal computer tra un ampio numero di sviluppatori software localizzati in diversi contesti territoriali e istituzionali. Il segreto delle comunità *open source*, a nostro parere, risiede nell'aver sviluppato un contesto formativo che favorisce la socializzazione e la condivisione della conoscenza su scala globale.⁶ Internet, in questa prospettiva, è solo il mezzo. Il vero valore aggiunto sta nella struttura e nell'organizzazione del contesto. Il cuore di questo contesto è dato da due fattori principali: l'istituzionalizzazione della pratica del *copy left* e un modello di sviluppo cooperativo. L'istituzione del *copy left* ha fornito il quadro "legislativo" di riferimento per stimolare la cooperazione e la condivisione nello sviluppo del software. Il modello di sviluppo ha, invece, canalizzato il potenziale intellettuale presente dal lato utente e distribuito a livello mondiale. Questo modello presenta similitudini forti con il modello distrettuale tipico del tessuto industriale italiano. Studiare il modello *open source* può rappresentare, quindi, un'opportunità per ri-immaginare i distretti industriali all'inizio del terzo millennio.

6. Per un approfondimento sul concetto di contesto formativo si veda Ciborra C., Lanzara G.F. (a cura di), *Labirinti dell'innovazione*, Etas, Milano 1999.

