

# Open Source e modelli di business: come creare valore per le PPAA. Il system integrator ed il caso Engineering

Gabriele Ruffatti  
Engineering Ingegneria Informatica  
gabriele.ruffatti@eng.it

## Keywords

Open Source, Modelli di Business, Enterprise Architecture, System Integrator, Ecologia del valore

## SOMMARIO

L'open source è un ottimo esempio da utilizzare per descrivere il business come ecologia, come luogo di produzione e di sviluppo della creatività e di valori condivisi: un'ecologia del valore. La stessa impresa moderna appartiene ad un eco-sistema, un sistema di network, un hypernetwork che ne influenza le prestazioni a breve o le performances di lungo, mutandone le forme organizzative e decisionali e condizionando gli stessi rapporti tra ciò che è deliberato e ciò che è emergente<sup>1</sup>. L'open source favorisce quindi l'instaurarsi di un ecosistema dove il modello di business del system integrator, centrato sulla realizzazione di progetti informatici, incontra le esigenze della Pubblica Amministrazione, intesa come azienda di servizio non priva di un proprio modello di business. Si può quindi instaurare un rapporto sinergico che crea valore per le Pubbliche Amministrazioni se si creano alcune condizioni di partenza e se si stimola lo sviluppo.

A partire da alcune assunzioni di base, quali la centralità del business quale elemento guida per l'evoluzione del fenomeno open source ed una precisa collocazione di tale software nell'ambito dello sviluppo di componenti di sistema e di applicazioni verticali, il testo individua alcuni modelli di business di chi sviluppa software open source e il loro rapporto rispetto ai diversi modelli d'uso.

A partire dall'esperienza diretta di Engineering Ingegneria Informatica, viene proposto un nuovo modello di business "centrato sul progetto", proprio del system integrator, e vengono esaminate le sue relazioni con le esigenze delle Pubbliche Amministrazioni.

Vengono inoltre proposte alcune indicazioni su come creare le condizioni di partenza e stimolare un rapporto sinergico tra integratore e Pubblica Amministrazione volto ad innescare e sostenere un'effettiva ecologia di valore attorno alla Pubblica Amministrazione italiana.

---

<sup>1</sup> I concetti di ecologia del valore e di eco-sistema d'impresa sono tratti dai lavori di Pilotti e Ganzaroli citati tra i riferimenti.

## CENTRALITA' DEL BUSINESS

Il business è centrale in tutte le attività umane. Hillman [6] sostiene che *"la psicologia del business non riguarda solo gli uomini d'affari, soltanto coloro che sono concretamente impegnati nell'industria, nel commercio, nell'economia. Il business rappresenta la ragione principale per cui tutti noi ci alziamo la mattina e il principio organizzatore di ogni nostra giornata. Occuparsi della propria giornata significa occuparsi del proprio business. Il business fornisce le idee che danno forma alla nostra vita, al suo successo, ai suoi valori, alle sue ambizioni"*. E ancora *"Il business non è semplicemente un fattore, una componente fra le molte che influiscono sulla nostra vita. Le sue idee costituiscono la trama e l'ordito fondamentali e imprescindibili su cui sono tessuti i modelli dei nostri comportamenti"*. In questo caso, si parla di psicologia dell'individuo e della società, di economia, di potere.

Una possibile definizione di business, tratta da wikipedia, è la seguente: *"In economia, il business è la scienza sociale che si occupa di gestire le persone per organizzare e mantenere una produttività collettiva volta a raggiungere specifici obiettivi creativi e produttivi, generalmente allo scopo di generare profitto"*<sup>2</sup>. Ancora una volta si parla di economia e di profitto, ma rimane il concetto di società.

Un'ulteriore definizione, di Osterwalder, Pigneur e Tucci [11] riguarda il modello di business: *"Uno strumento concettuale che contiene un elevato insieme di elementi e le loro relazioni e consente di esprimere la logica di business di una specifica organizzazione. E' una descrizione del valore che un'azienda offre ad uno o diversi segmenti di clientela, dell'architettura dell'azienda e della sua rete di partner che crea, commercializza e distribuisce tale valore e il capitale delle relazioni per generare flussi di ricavo profittevoli e sostenibili"*.

In quest'ultimo caso si parla ancora di profitto, ma si introducono, fra gli altri, i concetti di relazione, rete di partner, sostenibilità che, come vedremo, entrano pesantemente in gioco nel contesto open source. Se sostituiamo al *profitto* il termine *valore*, si apre una nuova prospettiva dove affermare, con Pilotti e Ganzaroli [13] che nel contesto di un'ecologia del

---

<sup>2</sup> Traduzione dell'autore da wikipedia:  
<http://en.wikipedia.org/wiki/Business>

valore: *“Il valore è il prodotto dell’interazione complessa tra una rete di soggetti che sono diversi tra loro, ma accomunati dalla condivisione di un comune spazio decisionale su cui detengono un comune interesse e a cui decidono di partecipare, apportando il proprio contributo che può essere più o meno volontario. Il valore, in questa prospettiva, non è riconducibile alla semplice somma del valore dei contributi apportati dai singoli, ma in parte è contenuto nella rete esperienziale e valoriale che lega gli attori che hanno partecipato volontariamente ed attivamente alla sua creazione”*.

Date queste definizioni, è possibile introdurre il tema di questo testo, individuandone poi la chiave di lettura.

Il software libero/open source<sup>3</sup> è ormai da tempo considerato una realtà affermata; secondo Golden [5]: *“Il software open source è mainstream...Il software open source è cresciuto nell’intero software stack... Se non hai ancora considerato l’opportunità di utilizzare l’open source, è questo il momento di farlo. I tuoi partner e concorrenti stanno implementando l’open source oggi. Perché non ti unisci a loro?”*. Per conferma, Driver, analista Gartner afferma che [2] *“Il software open source è un catalizzatore che ristrutturerà l’industria, producendo software di più alta qualità ad un costo più basso. Non distruggerà giganti come IBM e Microsoft, ma rivoluzionerà i mercati software spostando i flussi di ricavo verso i servizi ed il supporto piuttosto che sulle licenze”* e prevede che nel 2010 i prodotti open source costituiranno non più del 10% del intero portafoglio software delle aziende Global 2000<sup>4</sup> e che queste aziende valuteranno l’opportunità di adottare prodotti open source nell’80% dei propri investimenti software infrastrutturali e nel 25% dei propri investimenti software di business.

L’ambito di applicazione del software open source è sempre più ampio, ed è sempre più evidente la sua trasformazione che comprende anche una evoluzione verso modelli di sviluppo, sostegno e adozione da parte di aziende e di grandi organizzazioni, più che da parte delle comunità degli individui, come analizzato da Fitzgerald [3], che ha appositamente coniato il termine OSS 2.0.

Il software open source va analizzato, in prospettiva di maturità e sostenibilità, rispetto a precisi modelli di business e soprattutto rispetto ad un modello di business che non è solo quello dell’azienda che lo sviluppa e lo sostiene, ma che, secondo Ganzaroli e Pilotti [4], si inserisce in un più ampio *“ecosistema di attori che cooperano – combinando dinamicamente ed in modo innovativo cooperazione e competizione, egoismo e altruismo – alla produzione di un comune bagaglio di significati e, quindi, di valore/i”*.

<sup>3</sup> E’ sempre opportuno sottolineare che nel seguito si utilizzerà il termine open source con un’accezione del tutto generale. Per una definizione di software libero (free software) si veda <http://www.fsf.org/philosophy/free-sw.html>, mentre per una definizione di open source si veda <http://www.opensource.org/docs/definition.php>

<sup>4</sup> Per una definizione di aziende Global 2000 si veda <http://www.forbes.com/lists/2005/03/30/05f2000land.html> e

In questo ambito un system integrator, che ha come strumento di lavoro la propria competenza IT, non considera più l’open source solo un’opportunità, ma entra come attore fondamentale in un contesto di “ecologia del valore” di cui beneficiano sia lui che il proprio committente/utente, se si attivano alcune condizioni chiave:

- la connotazione infrastrutturale, aperta ed evolutiva del software
- un modello di business chiaro, aperto e sostenibile
- la coincidenza di visione e la chiarezza in alcune precise esigenze (riuso, crescita, sostenibilità) con il committente/utente.

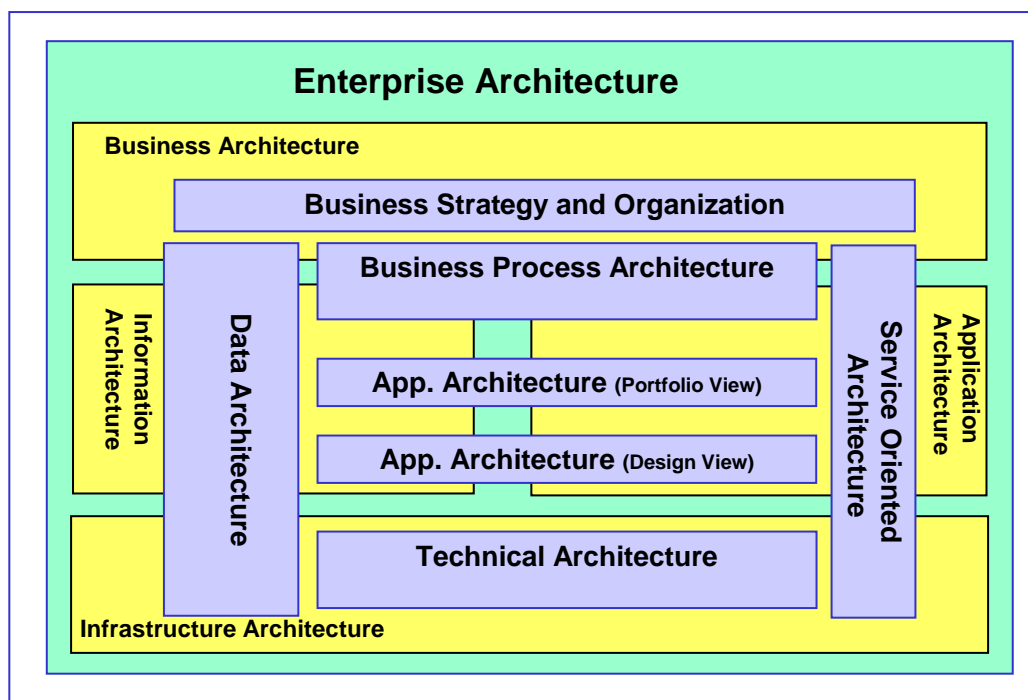
Quando il committente/utente è la Pubblica Amministrazione, organizzazione per la quale valgono i medesimi concetti di “business” e di “modello di business” già descritti, ma nel contesto di profitto legato ad un’industria di servizio, la cui efficienza deve essere giudicata anche, e soprattutto, in riferimento ai servizi che fornisce, ecco che diventa possibile instaurare un rapporto sinergico tra l’Integratore e l’Amministrazione, innescando così un’ecologia del valore.

## COLLOCAZIONE DEL SOFTWARE OPEN SOURCE

Un tema da chiarire in premessa riguarda la tipologia di software open source esaminata in questo contesto e l’ambito in cui si deve collocare per rispondere alle esigenze delle Pubbliche Amministrazioni. Nella presentazione del Convegno Pubblica Amministrazione Aperta e Libera PAAL 2007 di Pula (Cagliari), Marchesi [10] classifica in quattro categorie le tipologie di soluzioni open source oggi disponibili: software di sistema; automazione d’ufficio; componenti di sistema più grandi; applicazioni verticali.

Il software di sistema e di automazione d’ufficio è maturo, così come sono disponibili ottimi componenti per la realizzazione di sistemi più grandi (normalmente i sistemi informativi specifici offerti in risposta a bandi pubblici); la presenza di software open source in domini verticali sta rapidamente aumentando ed è comunque quest’ultimo in prospettiva l’ambito di maggior interesse per le Pubbliche Amministrazioni.

Nel contesto di questa trattazione, agli ambiti “componenti” e “applicazioni verticali” viene aggiunto quello che si può definire “middleware di fascia alta” o “middleware up-the-stack”: framework o piattaforme progettate e realizzate sia per rendere disponibili funzionalità e servizi agli utenti finali, che per essere adattate alle loro necessità. Grazie ad una progettazione coerente e all’adozione di standard aperti, essi consentono di integrare ulteriori componenti o applicativi verticali, sia open source che proprietari, facilitando il riuso e/o l’integrazione di ambienti esistenti, aumentando così anche il valore delle applicazioni già in uso. Nell’intento di ricondurre il tutto ad una schematizzazione più ampia, si può far riferimento al modello di Enterprise Architecture (fig. 1) proposto da Thorn [16].



**Fig.1 Enterprise Architecture**

L' Enterprise Architecture comprende la visione, i principi, gli standard, i processi che guidano l'acquisizione, la progettazione e lo sviluppo della tecnologia in un'azienda. Descrive le relazioni tra i processi di business, le informazioni, le applicazioni e la sottostante infrastruttura e fornisce un ambiente gestito per l'introduzione di nuova tecnologia. Thorn sintetizza così il valore dato da questa impostazione:

- consente l'allineamento dell'ICT con il modello e la strategia di business dell'organizzazione,
- facilita i cambiamenti sulla base di opportunità di business basate sulla tecnologia,
- facilita l'introduzione di nuove tecnologie,
- consente la standardizzazione,
- avvia il consolidamento dell'architettura informativa,
- riduce la complessità dell'integrazione delle applicazioni aziendali,
- facilita l'outsourcing, quando richiesto,
- ottimizza l'uso dei singoli elementi applicativi,
- riduce l'impatto di eventuali cambiamenti,
- riduce il time-to-market.

Oggi tale impostazione non è generalmente presente in aziende e Amministrazioni, ma può comunque essere considerata come modello di riferimento o punto di arrivo per una nuova organizzazione.

Se consideriamo quindi l'architettura di un'azienda o di un ente come un sistema rivolto al proprio business, che coinvolge strategie, organizzazione, processi, sistemi informativi e tecnologia, dove l'ICT non è al servizio del business, ma dove ICT e business seguono percorsi paralleli e sinergici, ecco che l'open source può ricoprire un ruolo strategico. Qualora posto a livello di architettura infrastrutturale e di architettura tecnica ha la potenzialità di imporre un modello tecnologicamente aperto e interoperabile, garantendo la

coesistenza negli strati applicativi sia di software open source che di soluzioni proprietarie. Consente inoltre di mantenere il controllo sull'innovazione della propria architettura anche come elemento di negoziazione nel forzare eventuali soluzioni applicative proprietarie ad adottare standard aperti. Va considerato poi che oggi l'open source inizia a dare valore anche in ambito di architetture dati e di architetture a servizi (SOA) e, sempre più nel futuro, in ambito di architettura applicativa.

Da un punto di vista meno tecnologico e più rivolto al processo di sviluppo, l'open source deve essere ormai considerato nell'ambito della "nuova generazione di progetti OS", caratterizzati da:

- approccio olistico e sviluppo pianificato,
- progettazione complessa di soluzioni in domini verticali per rispondere a requisiti di business,
- processo di sviluppo maturo,
- coinvolgimento non solo della comunità di sviluppatori/individui, ma di organizzazioni, aziende, consulenti e sempre più di utenti,
- servizi di supporto forniti da aziende e, in prospettiva, da network di aziende che ne garantiscono l'integrazione e la sostenibilità nel tempo.

Questa evoluzione, sia tecnologica che di processo, prosegue secondo una direzione che ancora una volta non porterà ad individuare l'open source come la soluzione "migliore" in ogni caso, ma che consentirà nel tempo di consolidare un nuovo ecosistema aperto e duraturo, basato sullo sviluppo collaborativo, all'interno del quale ogni partecipante, nel perseguire la propria strategia, contribuirà alla sostenibilità complessiva di un ampio sistema di soluzioni di "fascia alta" che nessuna azienda da sola sarebbe in grado di sviluppare.

## MODELLI DI BUSINESS OPEN SOURCE

Se il business è centrale nelle attività individuali e sociali, lo è di sicuro anche nella realizzazione e nella scelta di soluzioni open source da adottare. Il software open source non è “tutto uguale”: oltre alla comune esperienza, vi sono analisi sui progetti censiti in SourceForge, uno dei più noti e ampi repository open source (140.000 circa in gennaio 2007), come quelle di Weiss [19] e Klineciewicz [7], che dimostrano che non tutti i progetti sono attivi e, comunque, “uguali”. E soprattutto i modelli di business nell’open source “non sono tutti uguali”. E’ importante quindi introdurre questo ulteriore aspetto di valutazione utilizzando qualche schematizzazione.

Koenig [8] ha proposto il seguente elenco di modelli di business che indicano la strategia di chi sviluppa il software open source.

- *Patrocinio (patronage)*: spingere la “commoditizzazione” di una soluzione. L’obiettivo in questo caso non è il profitto, ma “sostenere” l’adozione di software open source per diversi motivi: accelerare l’adozione di uno standard, trasferire nel mercato i risultati della ricerca, bilanciare il ruolo dominante di un protagonista nel mercato.
- *Ottimizzazione*: spingere il risparmio dei costi per vendere valore aggiunto, in genere specifiche soluzioni proprietarie ottimizzate.
- *Doppia licenza (dual licensing)*: è uno schema cosiddetto ibrido che prevede sia il rilascio del software come licenza libera (spesso la GNU GPL), che la vendita dello stesso su licenza “proprietaria”, secondo lo schema EULA (End User License Agreement), affinché l’acquirente non ricada nelle obbligazioni di reciprocità della licenza libera. La versione di software distribuito con licenza libera e proprietaria può essere la stessa, ma vi sono soluzioni dove il software libero contiene meno funzionalità rispetto ad una versione “enterprise” o “professional” a pagamento che si rende necessaria per la realizzazione di progetti industriali; in quest’ultimo caso l’open source è una caratteristica puramente promozionale.
- *Consulenza*: l’open source è utilizzato per vendere servizi complementari, quali formazione, certificazione, supporto, integrazione.
- *Sottoscrizione (Subscription)*: i ricavi derivano da servizi ricorrenti quali il packaging, il bundling (integrazioni di soluzioni open source in uno specifico stack), la manutenzione.
- *Hosted*: la vendita di servizi (software as a service) basati su soluzioni open source.
- *Embedded*: prevede l’inclusione di software open source in hardware o altro software complesso; costituisce una leva per vendere una soluzione proprietaria.

Questi modelli, così come esposti, sono commentati e contestualizzati da Letellier [9], che aggiunge il seguente:

- *Bait and Hook*: l’open source è utilizzato per distribuire tecnologia su cui vendere complementi (software o servizi) allo scopo di ottenere un vantaggio competitivo. Per utilizzarlo con successo nella vendita di prodotti complementari proprietari è “opportuno” non adottare open standard; in questo caso, come per il dual licensing, l’open source è una caratteristica promozionale.

Letellier aggiunge ulteriori punti di vista dati dalla finestra di opportunità in cui sono inseriti i diversi modelli, che si affermano o cambiano su base temporale (opportunità temporanea, di primo avvio o di ultimo periodo), ma soprattutto aggiunge il punto di vista dell’utente, con i suoi modelli d’uso:

- *Riuso*: utilizzo di software esistente a minor costo, rispetto al software proprietario, eventualmente per utilizzi episodici, anche con finalità di ricerca di una maggior qualità.
- *Doppio sorgente*: alternativa o sostituzione totale o parziale di software proprietario, alla ricerca di minore dipendenza o per forzare il fornitore di software proprietario ad una riduzione di prezzi.
- *Outsourcing*: utilizzo della comunità open source come terza-parte cui delegare l’arricchimento e la manutenzione del software, con la volontà di rimanere sulla cresta della tecnologia, sfidare i concorrenti e sbarazzarsi di ciò che non costituisce più vantaggio competitivo.
- *R&D condivisa*: sviluppo condiviso con la comunità considerata al centro del processo di innovazione.

Ecco quindi che l’adozione del software open source è influenzata non solo da un modello di business (quello del produttore), ma dall’incontro di almeno due modelli di business: quello dell’azienda che lo produce e quello di chi lo adotta che, nel caso in esame, sono rispettivamente il system integrator e la Pubblica Amministrazione. Come vedremo, l’integratore che sviluppa e gestisce soluzioni open source propone un nuovo modello di business che unisce diversi aspetti tra i modelli dianzi elencati e che potenzialmente favorisce l’incontro con le esigenze della Pubblica Amministrazione italiana e, in buona misura, europea.

## IL RUOLO DEL SYSTEM INTEGRATOR

In un precedente lavoro di Ruffatti e Cazzin [15] sono stati schematizzati diversi punti di interesse di un system integrator nell’adozione e realizzazione di software open source:

- motivi economici, quali la riduzione dei costi di sviluppo, per realizzare software *good-enough* e software “adatto” al giusto costo; la riduzione del *time-to-market*; l’incremento della quota di mercato di software custom rispetto ai prodotti; la possibilità di estendere lo spettro dei servizi professionali (consulenza, manutenzione e

assistenza); la riduzione della dipendenza della manutenzione del software applicativo dal software di base; la possibilità di utilizzare la leva dei prodotti gratuiti per venderne altri a pagamento; nuove sinergie con enti, clienti, utenti, aziende e comunità di ricerca;

- motivi tecnologici, quali la collaborazione per risolvere l'aumento della complessità del software; l'utilizzo di standard aperti; l'incremento degli skill tecnici; l'aumento della flessibilità; la possibilità di sviluppare maggiore qualità; uno stimolo all'innovazione.

Per un system integrator l'open source è un'occasione importante per consentirgli di fare al meglio il mestiere che più conosce: realizzare soluzioni che soddisfino i requisiti dei propri clienti. Si tratta di mettere in campo le proprie competenze con alcune caratteristiche precise:

- la conoscenza del mercato e della tecnologia,
- la rapida adattabilità a nuovi contesti,
- l'indipendenza e la neutralità rispetto a fornitori e soluzioni (proprietarie e open source),
- la grande libertà di manovra e rapidità decisionale,
- l'elevata focalizzazione sul business,
- la visione imprenditoriale e la capacità manageriale,
- l'attitudine alla crescita sostenibile, in termini di rapporto costi/benefici.

Tali caratteristiche consentono di corrispondere ai principali bisogni delle imprese e delle amministrazioni committenti, quali:

- costi infrastrutturali di ingresso ridotti a zero e riduzione di sprechi,
- applicazioni realizzate su misura, aperte, facilmente adattabili e riusabili,
- disponibilità di un ampio spettro di servizi di supporto.

Viele [18], Direttore Ricerca e Innovazione di Engineering, nel 2005 affermava che *“Per un system integrator l'open source rappresenta più un'opportunità che un pericolo. Oggi non è possibile stabilire il valore potenziale del mercato stimolato dall'open source, ma le caratteristiche del fenomeno sono tali da far presumere una crescita progressiva nei prossimi anni. La sfida per un system integrator è la preparazione d'eccellenza, la sola caratteristica che può consentire di cogliere appieno i frutti di tale rivoluzione”*.

## **IL MODELLO DI BUSINESS DEL SYSTEM INTEGRATOR**

Il modello di business di un system integrator è classicamente individuato come modello di *consulenza* (consulting), rivolto alla vendita di formazione, certificazione, supporto, integrazione. Nella realtà, per un integratore che fonda almeno parte del suo business sull'open source, sia utilizzando soluzioni esistenti che sviluppandone e sostenendone di nuove, il modello è più complesso e in qualche modo attinge ad alcuni degli schemi elencati in precedenza, con l'esclusione di

quelli che vedono l'open source solo come opportunità promozionale.

E' quindi presente una componente di *patrocinio*, ad esempio quando l'integratore intende spingere la “commoditizzazione” di una soluzione, sia per le motivazioni proprie di tale modello (affermare l'adozione di uno standard, trasferire nel mercato risultati della ricerca, bilanciare il ruolo dominante di un protagonista nel mercato), che per evitare il lock-in indotto dai fornitori delle soluzioni proprietarie. Quest'ultimo non incide solo sull'utente finale, ma anche sull'integratore imponendo per l'applicazione da questi sviluppata costi di licenza e manutenzione, costi indotti dall'acquisizione di specifico hardware e/o software, grado di apertura definito, evoluzione condizionata dalle scelte del fornitore a costi imposti.

Nel modello dell'integratore è presente anche una componente rivolta alla vendita di servizi a valore aggiunto: *ottimizzazione e sottoscrizione*.

Ma più propriamente tale modello può essere così definito:

- *Project centric o “centrato sul progetto”*: il fine è la realizzazione del progetto del committente per il quale la soluzione open source offre caratteristiche preferibili ad una soluzione proprietaria quali disponibilità, apertura, modificabilità, modularità, integrabilità, adattabilità, riusabilità e scalabilità. Il progetto vale molto di più della soluzione adottata.

Questo modello, se correttamente esercitato, prevede che sia il software open source utilizzato che quello sviluppato costituiscano un mezzo per la realizzazione di progetti applicativi che a loro volta sono un mezzo per la crescita del software open source stesso.

In altri termini:

- da un lato si sfruttano le caratteristiche dell'open source per soddisfare i requisiti dei committenti nel costruire nel miglior modo possibile un'applicazione che risulta sempre realizzata su misura e quindi “adatta”;
- contemporaneamente si restituiscono alla comunità nuovi requisiti, nuovi sviluppi, feed-back e contributi.

Il system integrator è l'elemento attivatore di questo rapporto sinergico e uno dei principali attori proprio perché questo modo di operare è indotto dal “patrimonio genetico” che lo contraddistingue.

Tale modello ha come conseguenza che la scelta delle piattaforme e dei componenti open source individuati per una realizzazione “centrata sul progetto” e anche l'approccio allo sviluppo seguano precise linee guida.

- L'impostazione progettuale deve essere coerente, con il riequilibrio dei pesi in gioco nello sviluppo: troppo spesso nella realizzazione di un progetto l'attenzione è posta solo sull'integrazione e l'adattamento di un prodotto. Di contro, una soluzione orientata al progetto è concepita per realizzare applicazioni adatte alle esigenze dell'utente, con l'attenzione particolarmente rivolta all'impostazione progettuale. Quando si utilizza un framework, una piattaforma o una soluzione progettata non solo per rendere disponibili funzionalità e servizi agli utenti finali, ma soprattutto per adattarli alle loro necessità, si

facilita il riuso e l'integrazione di ambienti esistenti, aumentando anche il valore delle applicazioni già in uso. In questo modo è anche agevolato l'avvio del progetto, grazie ad un diverso equilibrio tra dimensione dello sviluppo, costo e qualità dei risultati in confronto alle distorsioni economiche evidenti in diversi progetti realizzati a partire da prodotti proprietari o comunque "chiusi" in termini progettuali.

- Il processo di sviluppo è preferibile sia evolutivo o agile: un approccio evolutivo (o agile, se è ritenuto adatto al contesto) consente di evitare di avviare lo sviluppo di grandi progetti monolitici e può aiutare a raggiungere gli obiettivi progettuali a partire da primi sviluppi, "realizzati in piccolo, ma progettati in grande", conseguendo subito i primi risultati.
- In alcuni casi è opportuno attivare una politica di sperimentazione: molte aziende o amministrazioni non si sentono ancora a proprio agio con l'open source. Cercano buone funzionalità e un'usabilità accattivante, ma nella realtà necessitano di una soluzione che sia realmente adattabile alle proprie esigenze. Temono di non trovare né una soluzione di livello *enterprise* (di qualità, robusta, scalabile), né un vero supporto (installazione/configurazione, correzione di errori, manutenzione, formazione, garanzia). L'avvio di un primo progetto, ad un ragionevole livello di costo (risparmiando comunque sui costi di licenza), consente di aumentare sia la confidenza con le soluzioni open source, che la conoscenza dell'organizzazione interna sulle tematiche innescate dall'utilizzo di questa tipologia di soluzioni.

Da quanto detto appare evidente che il modello di business *centrato sul progetto* è potenzialmente efficace sia per chi lo esercita (l'integratore) che per chi lo "subisce" (il committente, nel nostro caso la Pubblica Amministrazione), anche se alcuni punti restano aperti. Per introdurli può essere utile un confronto con il modello *a doppia licenza* che si sta affermando sempre di più tra le aziende che sviluppano soluzioni open source.

Il modello dual licensing:

- prevede la vendita di licenze software (generalmente a costo inferiore delle analoghe soluzioni proprietarie) e servizi di sottoscrizione;
- è capace di costruire interesse e significative comunità di utenti (esempi: eXoPlatform, Funambol e Alfresco, oltre ai classici Sleepycat, MySQL e TrollTech), ma è sovente guidato da un'azienda che ne detiene uno stretto controllo;
- è misurabile e consente di realizzare un business plan sulla base di costi e ricavi nel tempo (dall'analisi di Valimaki [17], condotta su tre casi significativi, emerge come il principale ritorno economico provenga dalla vendita di licenze, che generalmente supera il 50% dei ricavi)
- investe quote anche significative di ricavi in attività di marketing;
- spesso è sostenuto da investitori istituzionali (*venture capitalist*): si vedano a questo proposito le considerazioni di Wheeler [20] dove appare

evidente un elevato interesse e un significativo tasso di crescita delle *venture capital firms* che finanziano startup impegnate nell'open source "commerciale" (incremento del 131% nel 2006 con proposte di investimento che dal 2005 a 2006 sono aumentate di due o tre volte).

Il modello a doppia licenza viene esercitato in modi diversi (un'interessante posizione è quella espressa da Capobianco [1]), prevede sempre il coinvolgimento della comunità e può quindi essere considerato open source a tutti gli effetti, anche se inizia ad essere percepito come una semplice particolarità del modello proprietario con costi di licenza inferiori. Alcuni elementi, quali la presenza di investitori istituzionali e non industriali, che si aspettano ritorni economici di breve-medio periodo, il controllo totale da parte dell'azienda che sviluppa la soluzione, le caratteristiche della soluzione stessa generalmente tipiche del prodotto più che della piattaforma, soprattutto in presenza di differenze tra la versione libera e quella a pagamento, fanno apparire questo modello come più adatto ad un mercato IT d'oltreoceano generalmente "guidato" da aziende di prodotto e servizio.

Il mercato IT italiano e generalmente europeo è invece "guidato" da software house e integratori più rivolti alla realizzazione di progetti su misura. Ecco che nell'ambito della realizzazione di soluzioni open source il modello *project centric* si denota come potenzialmente più adatto, con diverse caratteristiche ma anche alcune debolezze:

- non prevede la vendita di licenze software, ma eventualmente la vendita di servizi e di sottoscrizione;
- incontra maggiori difficoltà nel costruire interesse e significative comunità di utenti, sia perché è comunque guidato da un'azienda che ne gestisce lo sviluppo, almeno finché la comunità dei contributori non cresce significativamente, sia perché una soluzione non centrata sul prodotto appare meno "attraente" quando confrontata con analoghe soluzioni proprietarie o "a doppia licenza";
- investe generalmente quote ridotte in attività di marketing, in misura limitata a sostenere la crescita della reputazione della soluzione;
- è sostenuto dalla stessa azienda, che a prescindere dalla compagine dei suoi finanziatori, generalmente non investe solo nell'open source, ma vede questo come un'opportunità aggiuntiva per mantenere ed incrementare il business tradizionale;
- non è misurabile: le quote derivanti dalla sottoscrizione e dalla consulenza (formazione, supporto all'avvio) appaiono non essere significative, mentre il principale ritorno economico che deriva dalla realizzazione di progetti applicativi non può essere quantificato come ritorno diretto della realizzazione della soluzione aperta sviluppata ed utilizzata (spesso neanche l'assegnazione del progetto all'integratore può essere attribuita allo sviluppo della soluzione open source).



Una criticità di questo modello, considerazione che però vale e forse ancora di più per quello a doppia licenza, è la presenza di un'azienda "forte" che ne governa lo sviluppo. Dicono Ganzaroli e Pilotti [4]: *"Il modello di funzionamento delle comunità OS si basa su un rapporto di fiducia che riesce ad attivare e stimolare continuamente la produzione di nuova fiducia. I fattori determinati di tale rapporto si possono schematizzare nella strategia di ingaggio e mobilitazione (partecipazione nella produzione del software, rilascio continuo di nuove versioni), nei sistemi di pianificazione e controllo utilizzati (rivolti non solo ad una efficienza produttiva, statica, ma anche ad una efficienza dinamica nella costruzione di risorse relazionali e fiduciarie), nel gioco della reputazione in cui la comunità tende a mantenere nel tempo il valore della propria reputazione e, da ultimo, la sostenibilità del processo legato alla natura distribuita delle comunità". "Il meccanismo di creazione e riproduzione della fiducia si può rompere nel momento in cui si vengono a creare delle forti asimmetrie di potere legate a posizioni di rendita. Queste situazioni possono emergere specialmente nel caso delle comunità dove si è deciso di incentivare la partecipazione da parte delle imprese private. In questo caso, infatti, imprese con un enorme potere di mercato potrebbero sfruttare la loro posizione per appropriarsi del valore del codice sorgente prodotto dalla comunità e spingere le imprese minori e gli individui ad abbandonare la comunità"*.

Si è già sottolineato, quando si è analizzato il modello di business a doppia licenza, come la presenza forte di un'azienda nella governance di un progetto open source costituisca un elemento critico, superabile, ma non completamente, solo quando il progetto riveste caratteristiche particolari di novità o efficacia. Anche nel modello *project centric* è presente il rischio che la comunità cresca solo intorno agli utilizzatori del software, limitando però il ruolo degli agenti attivi nel sostegno e nella conduzione del progetto, se si escludono l'azienda guida e le altre aziende operanti nell'open source che con lei collaborano.

Un'ulteriore criticità è data dalla difficoltà di ottenere una misura del modello "centrato sul progetto", il che può portare ad una mancanza di credibilità e di affermazione. Il punto è che tale modello appartiene più ad una ecologia del valore che ad una dimensione economica, con la difficoltà intrinseca di valutare in modo quantitativo la prima e la necessità di mantenere comunque le due componenti in un equilibrio centrato sul business.

Ma la principale differenza tra il modello a doppia licenza e il modello centrato sul progetto rimane il tipo di soluzione prodotta; nel primo caso l'evoluzione è guidata dal produttore, dove la comunità, qualora affermata, agisce come forza di persuasione e negoziazione; nel secondo caso l'evoluzione è governata dall'integratore, ma è guidata, in modo diretto o indiretto, dai progetti e quindi dai loro committenti e dagli utenti.

## LE ESIGENZE DELLE PUBBLICHE AMMINISTRAZIONI

Le esigenze delle Pubbliche Amministrazioni nell'ambito dell'informatizzazione sono molteplici e ampiamente dibattute. Una breve sintesi delle opportunità offerte dall'open source è in [15], dove per le pubbliche amministrazioni si individuano sia motivazioni economiche di base, quali la riduzione dei costi di acquisizione per ottenere software "good-enough" (prodotti) e "software adatto" (applicazioni) al giusto prezzo grazie anche all'abbattimento del costo delle licenze, la riduzione degli sprechi (value for money), la riduzione del *time-to-market*, l'indipendenza dai fornitori con un aumento della sostituibilità degli stessi, la riduzione del TCO (*Total Cost of Ownership*); che motivi tecnologici, volti all'adozione di standard aperti, all'aumento di flessibilità e qualità. A questi si possono aggiungere l'opportunità di favorire il riuso e l'interoperabilità; la velocità nella produzione di nuovi rilasci, dove il piano di realizzazione è guidato dall'Amministrazione committente e non dalle strategie del fornitore; la possibilità di utilizzare soluzioni localizzate per il mercato specifico ed infine una più elevata ispezionabilità.

Allargando lo spettro al "sistema Paese" si possono individuare anche l'opportunità di favorire il pubblico accesso, la diffusione e la libertà nell'utilizzo dell'informazione; la possibilità di frenare la creazione di monopoli; di liberare risorse economiche impegnate per l'acquisizione di software su licenza per dirottarle verso lo sviluppo di "software custom"; l'opportunità di indirizzare finanziamenti al mercato nazionale ed europeo allo scopo di rivedere gli equilibri di mercato (anche attraverso la negoziazione); lo stimolo della competitività; lo sviluppo tecnologico e la nascita di nuovi soggetti e nuove forme di profitto; la possibilità di attivare una politica industriale del software in un contesto europeo e favorire nuove collaborazioni tra enti, clienti, utenti, aziende di diverse dimensioni ed attività di ricerca.

Ma poiché non rientra tra gli obiettivi di questo scritto approfondire questi argomenti, è sufficiente partire da alcune assunti e lasciar parlare l'attualità.

Le assunzioni di base (ampiamente dibattute, che non valgono per la generalità dell'open source, che come già detto non è "tutto uguale", ma che in molte situazioni sono verificate dalla realtà dei fatti) possono essere tratte dalla presentazione di Marchesi al Convegno PAAL 2007 di Pula (Cagliari) [10], dove si individuano alcuni vantaggi economici, tecnologici e sociali del software open source:

- è meno costoso;
- è di solito recente e tecnologicamente avanzato;
- è di qualità superiore, più sicuro e maggiormente conforme agli standard;
- facilita lo sviluppo di competenze informatiche locali.

In termini di attualità, il Ministro per le Riforme e l'Innovazione Luigi Nicolais ha recentemente definito le linee strategiche per realizzare il sistema italiano di e-government, sottolineando come la riforma della PA

debba essere il frutto di una "rivoluzione culturale per cui l'informatica non è più un fine ma un mezzo" [14]. Per tutti i sette obiettivi che saranno oggetto di direttiva ministeriale, qui elencati, la creazione di un ecosistema open source può essere l'elemento scatenante della crescita di un'ecologia di valore efficace:

- 1) maggiore efficienza della Pubblica Amministrazione, con l'attuazione del codice della PA digitale;
- 2) cooperazione tra le amministrazioni;
- 3) maggiore trasparenza nella spesa pubblica attraverso l'informatizzazione dei pagamenti e un maggiore ricorso all'e-procurement;
- 4) creazione di una cittadinanza digitale;
- 5) monitoraggio delle iniziative della PA, per misurarne l'efficienza;
- 6) promozione della competitività delle imprese;
- 7) lancio di iniziative per rendere l'Italia protagonista nel processo di innovazione in Europa.

Si tratta quindi di utilizzare le risorse IT per dare efficienza, garantire cooperazione e interoperabilità, realizzare e diffondere nuove soluzioni applicative aperte, promuovere l'e-democracy, realizzare infrastrutture che supportino la governance del sistema, il tutto stimolando la competitività delle imprese anche in un contesto sopranazionale ed europeo.

Considerando che comunque il primo passo è fornire alla Pubblica Amministrazione, che ricordiamo è un'azienda di servizio, applicazioni che rispondano ai requisiti richiesti, sempre diversi, ma con la possibilità di:

- essere riusabili;
- garantire interoperabilità tra le diverse applicazioni sia interne che tra le diverse Amministrazioni;
- essere efficaci sia verso l'utenza interna (l'organizzazione) che verso l'utenza esterna (i cittadini).

## IL CASO ENGINEERING

Il percorso che ha seguito Engineering Ingegneria Informatica ([www.eng.it](http://www.eng.it)) per entrare nell'ecosistema open source ha seguito precise linee guida.

Dopo una prima fase che ha visto l'azienda utilizzare soluzioni open source e supportare le comunità soprattutto nell'ambito dei progetti di ricerca europei, nel 2004 è arrivata la decisione di rilasciare e di supportare in ambito open source un software già maturo: il framework di sviluppo java Spago ([www.spagoworld.org](http://www.spagoworld.org)). Questo ha consentito non solo un diverso accreditamento, ma la possibilità di acquisire la necessaria esperienza sul campo per avviare una successiva fase a partire dalla decisione di avviare un nuovo progetto di sviluppo interamente open source, fin dalla sua organizzazione e progettazione: il sito di progetto è stato aperto prima di scrivere una sola riga di codice e sono state inizialmente pubblicate la visione e l'idea progettuale. Il progetto SpagoBI ([www.spagobi.org](http://www.spagobi.org)) è stato avviato con l'adozione di precise linee guida: a fianco dei temi "classici" posti in evidenza dal fenomeno open source (comunità di sviluppo e di pratica, processo di sviluppo, standard aperti, qualità, innovazione,

licenze), grande attenzione è stata rivolta alle motivazioni del mercato, sia economiche (riduzione dei costi, maggior spazio ai progetti di integrazione, aumento dello spettro dei servizi professionali, riduzione del *time-to-market*, sviluppo di soluzioni *custom* "adatte") che tecnologiche (riduzione della complessità del software, integrabilità ed interoperabilità, innovazione, flessibilità e qualità) e - perché no? - sociali e politiche: nuovi equilibri di mercato, maggiore competitività, incremento allo sviluppo tecnologico e all'innovazione, rilancio dell'industria nazionale ed europea, aumento della cooperazione.

Soprattutto Engineering ha inteso posizionarsi sul mercato open source come *system integrator* già indipendente rispetto alle soluzioni proprietarie, intenzionato a mantenere laicità di approccio anche rispetto alle soluzioni open source, ma anche fortemente rivolto ad un uso industriale di tale software. In tale ambito è stato sviluppato un modello di *Enterprise Open Source* per quanto riguarda lo sviluppo di soluzioni, e con esse di progetti, per la Pubblica Amministrazione e le grandi imprese, riunendo le competenze acquisite sul "modello originale" open source alla "genetica" capacità progettuale e realizzativa aziendale, e parallelamente un modello di *Professional Open Source* per quanto riguarda il "supporto certificato" sia a tali soluzioni, che alle infrastrutture di base (sistemi operativi, application server, data base).

Le linee guida che hanno indirizzato questo percorso possono essere sintetizzate nel seguito.

- La diffusione delle soluzioni sviluppate e supportate, allo scopo di ottenere il *net-effect* essenziale per il successo di ogni progetto open source. In questa linea si inserisce la scelta di aderire al Consorzio ObjectWeb ([www.objectweb.org](http://www.objectweb.org)), un consorzio no-profit focalizzato sulle soluzioni di middleware a livello industriale, e di contribuirvi i due progetti Spago e SpagoBI.
- Il coinvolgimento di tutti i soggetti che accettino l'ingaggio perché vedono la possibilità di benefici comuni, dato che il risultato delle attività, essendo open source, rimane di "pubblico dominio": in questo senso si inseriscono i numerosi contatti avviati e le attività intraprese con università italiane ed esponenti del mondo accademico.
- La solidità del supporto alle soluzioni, attraverso una certificazione tecnologica, un supporto tecnico ed una "garanzia" legale, data dall'adozione di un preciso schema di licenze.
- L'avvio di alleanze e sinergie, soprattutto in ambito europeo, con altre aziende operanti nel mercato open source, sia per aumentare la massa critica, sia per raccogliere l'interesse internazionale che già oggi i paesi emergenti rivolgono al fenomeno open source sviluppato "da questa parte dell'oceano".
- Una sempre maggiore "focalizzazione" degli interventi verso iniziative industriali tramite il coinvolgimento non solo delle comunità di



sviluppo, ma soprattutto di progetti, di altre imprese e degli utenti.

- Il continuo sviluppo di nuove integrazioni per accogliere le migliori soluzioni disponibili in un percorso di innovazione continua.
- La volontà di non eludere i temi critici all'interno del dibattito nazionale ed europeo con il coraggio di adottare posizioni ben definite; in questo ambito si può ascrivere l'opposizione alla proposta di direttiva europea sulla brevettabilità del software, l'attività in favore di una piena interoperabilità mediante l'adozione di standard aperti e, ove possibile, la spinta all'adozione di software open source nella Pubblica Amministrazione.

L'elemento fondamentale per il successo delle soluzioni open source di Engineering è stato la scelta di non agire da soli, ma di collegarsi ad una ampia comunità internazionale che desse particolare attenzione alle necessità degli utenti finali e delle imprese, alla collaborazione tra aziende e organizzazioni già attive nell'ecosistema, senza tuttavia trascurare il ruolo della comunità degli sviluppatori e degli individui. Il passaggio dei progetti di Spago e SpagoBI, dall'ampio e conosciuto *repository* open source SourceForge, privo però di una precisa connotazione, al Forge del Consorzio ObjectWeb, ha segnato quindi una scelta precisa: quella di inserirsi in una comunità connotata da una precisa identità tesa a promuovere ed affermare un ecosistema sostenibile e duraturo centrato su soluzioni di *middleware* open source che possano aiutare sia le imprese appartenenti al consorzio a soddisfare i propri obiettivi di business, che le Amministrazioni e le imprese utenti a soddisfare i propri bisogni. Ciò ha reso evidente la strategia open source di Engineering tesa a condividere i progetti con la comunità e ad integrarli con altre soluzioni nella ricerca di tutte le opportunità possibili, per cui Spago e SpagoBI oggi non rappresentano solo la soluzione adatta per soddisfare ambiti ben precisi, quali rispettivamente lo sviluppo di applicazioni J2EE in ambienti SOA (Service Oriented Architecture) o la realizzazione di progetti di Business Intelligence, ma sono anche un pezzo del più ampio stack di *middleware* di ObjectWeb. Tale strategia persegue l'obiettivo ulteriore di consolidare tali soluzioni a livello industriale, di stabilire collaborazioni a livello internazionale con particolare attenzione ai paesi emergenti che guardano con interesse all'Europa, sfruttando ad esempio la partnership consolidata tra ObjectWeb ed il consorzio asiatico OrientWare ([www.orientware.org](http://www.orientware.org)).

La partecipazione ad ObjectWeb è anche l'opportunità per condividere un percorso comune nel sostenere la trasformazione del Consorzio verso una nuova identità internazionale più forte e consolidata<sup>5</sup>, guidata da una "filosofia" aperta e volta a promuovere non solo soluzioni tecnologiche ma anche un nuovo modo di intendere il business. Il valore derivato da questo percorso è sicuramente elevato e non sempre monetizzabile, anche se emergono primi dati quantitativi provenienti dalla realizzazione di numerosi

progetti basati su soluzioni open source in Italia, da diverse proposte di realizzazione, tuttora in corso, presentate nel mercato europeo grazie alle partnership consolidate, che denotano non solo un'ulteriore spinta data dall'open source all'internazionalizzazione delle attività di Engineering, ma anche la penetrazione in nuovi settori di mercato e, non ultima, la creazione di un network di aziende che forniscono supporto a un intero *stack* open source, con la proposta ai clienti di un'interfaccia unica volta ad eliminare problemi di distanza, di lingua e di complessità nella gestione del servizio. Il tutto, al mero costo di mettere in atto una collaborazione aperta, di partecipare a sforzi di promozione comune e di impegnarsi in favore di scelte cruciali come quella contro la brevettabilità del software e a favore dell'innovazione, della competizione aperta e della libertà di impresa.

## COME ATTIVARE UN RAPPORTO SINERGICO

Secondo Ganzaroli e Pilotti [4] *"Lo studio dell'OS non ha valore solo se fine a se stesso, ma anche se letto alla luce del problema più ampio di stimolare la partecipazione dei cittadini alla creazione del bene pubblico. Il modello OS, da questo punto di vista, pare indicare una direzione, che sembra particolarmente appropriata alla produzione di beni pubblici di natura complessa. La direzione proposta, soprattutto nel caso dei beni complessi, è quella di una maggiore liberalizzazione non solo nella fase di commercializzazione di questi beni, ma, soprattutto, nelle condizioni di accesso ai fattori produttivi. E' necessario, in altre parole, investire nella creazione delle condizioni di ambiente che spingono larghe fette della cittadinanza ad investire, interagire ed associarsi nella produzione di questi beni."*

Ma le cose non sempre accadono da sole, bisogna creare le condizioni di partenza e stimolare lo sviluppo. Abbiamo visto come l'incontro tra il modello di business di un system integrator (definito come "centrato sul progetto") e le esigenze delle Pubbliche Amministrazioni può creare una sinergia volta ad innescare un'ecologia del valore. Si tratta di individuare quali sono le condizioni utili, o necessarie, per attivare tale rapporto.

### *Il ruolo imprenditoriale*

L'imprenditore - il system integrator - ha tutto l'interesse a stimolare la crescita di un tale ecosistema e, qualora entrato in tale ordine di idee, ne possiede le caratteristiche e le potenzialità come l'apertura ad operare in partnership sia con la committenza, anche tramite forme di co-investimento allo scopo di "fare sistema", che con altre aziende; possiede inoltre la capacità di accettare le sfide del mercato sia a livello nazionale che internazionale, dimensione necessaria quando si opera con l'open source. Quello che gli deve essere richiesto è la volontà di agire come attore in rete con altri attori di natura e dimensioni diverse, senza farsi prendere dalla tentazione di voler acquisire una posizione dominante.

<sup>5</sup> Il riferimento è al nuovo Consorzio OW2

### *Il ruolo di sostegno della Pubblica Amministrazione*

La Pubblica Amministrazione, come committente, riveste un ruolo fondamentale nel creare le condizioni di partenza. Diverse sono le opportunità e le possibilità, tra cui alcune già in essere:

- lo stimolo alla standardizzazione delle comunicazioni fra le diverse Amministrazioni;
- l'avvio di politiche di interoperabilità tra i sistemi, anche per ridurre la complessità dei procedimenti amministrativi;
- lo stimolo all'innovazione dei processi;
- la creazione di occasioni di network tra i diversi attori (committenza ed imprese) anche attraverso la consorzialità.

Le attività specifiche, volte a consolidare un reale ecosistema open source capace di sfruttare le potenzialità date dal sistema delle imprese, dovrebbero essere rivolte:

- alla creazione di un insieme di applicazioni open source per diversi ambiti applicativi,
- ad iniziative di governo che ne favoriscano il riuso e ne garantiscano la sostenibilità nel tempo.

Circa il primo punto va posta attenzione nella definizione di precisi criteri di predisposizione delle richieste d'offerta o di bandi pubblici dove si preveda anche il rilascio open source del software commissionato e, quando le condizioni lo consentono (dimensione dell'appalto), la sua gestione iniziale secondo le metodiche proprie dell'open source. Circa il secondo punto, l'attenzione dovrebbe essere rivolta a definire linee guida di sviluppo e metodi di assessment ed alla creazione di Forge<sup>6</sup> di supporto e di Centri di Competenza specifici per il sostegno e la diffusione delle soluzioni inserite in tali Forge.

### *Il ruolo del legislatore*

Il ruolo del Legislatore è quello di fornire il supporto normativo anche con iniziative che indichino l'open source come opportunità da perseguire con decisione - e non come scelta esclusiva - e che, senza scendere nella definizione di soluzioni tecnologiche, indichi un percorso che comprenda opportunità e stimoli rivolti, ad esempio:

- al finanziamento di soluzioni e di strumenti a supporto all'open source (metodi e soluzioni di governance, Forge, Centri di Competenza) per favorire la nascita delle già citate ecologie del valore,
- al finanziamento di alcuni progetti di sistema in ambiti specifici (scuola, sanità, sicurezza) inserendone lo sviluppo in un contesto open source,
- all'aggregazione del know-how scientifico (università e ricerca) e tecnologico (aziende),

---

<sup>6</sup> Per Forge si intende un repository di progetti open source che contempla tutti gli strumenti a supporto della pubblicazione, cooperazione e gestione del software quali portale di progetto, area di download, mailing lists, strumenti di sviluppo cooperativo. Il Forge è l'elemento di supporto alle comunità e in qualche modo ne diventa l'elemento distintivo (esempi sono quello di SourceForge, ObjectWeb, Apache, Savannah)

- allo stimolo al co-investimento ed alla consorzialità tra enti ed imprese.

## **CONCLUSIONI E PROBLEMI APERTI**

L'open source, come si è visto, è un ottimo esempio da utilizzare per descrivere il business come ecologia, come luogo di produzione e sviluppo della creatività e di valori condivisi: un'ecologia del valore. Citando Pilotti e Ganzaroli [12], la stessa impresa moderna *“appartiene ad un eco-sistema, un sistema di network, un hypernetwork che ne influenza le prestazioni a breve o le performances di lungo, mutandone le forme organizzative e decisionali e condizionando gli stessi rapporti tra ciò che è deliberato e ciò che è emergente”*. Il business è centrale nell'open source e l'indagine sui suoi modelli di riferimento rimane aperta in quanto essi stessi sono in continua evoluzione; in essi è presente l'elemento costante delle relazioni ed interconnessioni tra diversi attori in rete come sistemi che aggiungono valore in un rapporto reciproco di tipo *win-win*.

Il modello di business di un system integrator, centrato sulla realizzazione di progetti informatici, incontra le esigenze della Pubblica Amministrazione, intesa come azienda di servizio non priva di un proprio modello di business. Si può quindi instaurare un rapporto sinergico che crea valore per le Pubbliche Amministrazioni se si creano alcune condizioni di partenza e se si stimola lo sviluppo.

Una considerazione conclusiva derivante dall'esame del caso Engineering, concretizzato soprattutto nella realizzazione del progetto SpagoBI<sup>7</sup>, può iniziare con l'esame del trasferimento dei valori che sono alla base degli sviluppi open source su una realtà economica che necessariamente deve essere dinamica per adattarsi a situazioni in continua evoluzione. Tale analisi sarà oggetto di successivi lavori, ma si può già intravedere quale può essere il percorso che l'azienda dovrà seguire. Engineering dovrà periodicamente interrogarsi sul complesso sistema di reti cui appartiene e sugli agenti che a diversi livelli e titoli contribuiscono valore alle proprie attività, in un rapporto di mutuo sostegno, in modo da poter intravedere le proprie prestazioni di lungo periodo, pur sapendosi costantemente adattare ai mutati contesti che periodicamente influenzano le performance di breve. Ecco quindi che per un'azienda conoscere il proprio ecosistema e sostenere ed avvalorare i rapporti di network tra i diversi agenti diventa un'attività significativa da inserire in un piano industriale di crescita, al pari dell'individuare i mercati emergenti, le nuove soluzioni da proporre o la governance del modello produttivo.

Di pari passo, le Pubbliche Amministrazioni possiedono ancora più di un'impresa le motivazioni di base per divenire il motore propulsivo di un ecosistema

---

<sup>7</sup> Un'analisi che rappresenta Engineering come impresa ecologica sulla base del caso di studio del progetto SpagoBI è affrontata nel testo di Pilotti e Ganzaroli [12], in fase di pubblicazione.

del valore e possono attivarlo se inseriscono tale obiettivo nelle proprie linee strategiche.

Non tutto è però così semplice e restano alcuni aspetti da approfondire.

#### *Efficacia del modello “centrato sul progetto”*

Come si è visto, sono presenti due criticità nel modello *project centric*: la difficoltà di misurarlo e la crescita della comunità. La prima introduce il problema della possibile mancanza di credibilità e di affermazione. Il punto è che tale modello appartiene più ad una ecologia di valore che ad una dimensione economica, con la difficoltà intrinseca di valutare in modo quantitativo la prima e la necessità di mantenere comunque le due componenti in un equilibrio centrato sul business. La misura del modello, anche se in termini qualitativi, potrà essere oggetto di successive analisi, ma appare evidente che oggi le motivazioni alla base dello stesso sono più legate alla competitività che include reciprocità e condivisione più che all'iper-razionalismo ed alla iper-competizione.

La seconda criticità introduce un ulteriore aspetto:

#### *Il ruolo delle imprese nella comunità open source*

Sia nel modello di business a doppia licenza che nel modello *project centric* la presenza forte di un'azienda nella governance di un progetto open source costituisce un elemento critico, superabile, ma non completamente, solo quando il progetto riveste caratteristiche particolari di novità o efficacia. Anche nell'esperienza Engineering emerge come dato di fatto la difficoltà a far crescere e diffondere la comunità non tanto degli utilizzatori del software, quanto quella degli agenti attivi nel sostegno e nella guida del progetto, se si escludono le altre aziende operanti nell'open source che collaborano con Engineering. Non si ritiene che il tema del rapporto di fiducia tra comunità e azienda e tra aziende di diversa dimensione sia completamente esaurito, ben sapendo che rimane cruciale nel costruire una effettiva ecologia di valore soprattutto se centrata sulla Pubblica Amministrazione.

#### *La natura delle imprese IT nel contesto italiano*

Alcune difficoltà nascono dalla natura delle imprese IT nel contesto italiano: la scarsa presenza di aziende di rilevanti dimensioni, nei confronti di un elevato numero di aziende informatiche che non superano le dieci unità produttive; la italiana difficoltà alla consorzialità, alla condivisione e ad operare in rete; la diffidenza verso interlocutori forti, per il timore di subire posizioni dominanti, il che può portare ad escludere anziché includere esperienze e contributi; la perenne “confusione” che regna quando si affronta il tema dell'open source, dove si mescolano tematiche industriali a comunitarie, di business a valutazioni etiche, con una elevata presenza di produttori “proprietari” di hardware e software rispetto agli integratori ed ai produttori di servizi.

#### *Il ruolo di sostegno delle Pubbliche Amministrazioni*

Altre difficoltà nascono, probabilmente, dall'efficacia di un ruolo di sostegno esercitato dalle Pubbliche Amministrazioni. La presenza in ambito open source di iniziative diverse, non sempre coordinate, può costituire una ricchezza, ma di contro, può essere un

freno all'affermarsi dell'ecosistema delineato. La mancanza di iniziativa o iniziative troppo deboli (qualche volta uno “scatto in avanti” potrebbe essere opportuno, come nel caso della municipalità di Monaco di Baviera) oppure, dal lato opposto, di iniziative eccessivamente orientate, possono creare situazioni non condivise, con il rischio di ostacolare la crescita naturale del necessario retro-terra culturale di base.

#### *Il ruolo del Legislatore*

Lo stesso ruolo del Legislatore è di difficile lettura nel contesto nazionale dove talvolta è più facile, o più opportuno, non prendere posizione rispetto a tematiche cruciali o dove la difficoltà a trovare una sintesi tra posizioni non sempre condivise causa quantomeno lentezza nel favorire provvedimenti di supporto. Ciò nonostante, questo è un ambito in cui sarà necessario fare dei passi in avanti se si vuole favorire e sostenere efficacemente un ecosistema intorno alla Pubblica Amministrazione che non può da solo regolarsi e trovare i mezzi per sostenersi.

Tutti questi aspetti potranno essere affrontati nei dibattiti futuri e troveranno sicuramente opinioni discordanti. Nell'impossibilità di individuare regole certe, la direzione proposta è quella di iniziare con il valorizzare alcune esperienze di riferimento e costruire intorno ad esse dei primi nuclei operativi.

## RIFERIMENTI

[1] Capobianco, F. *My Honest Dual Licensing*, blog, July 2006

<http://www.funambol.com/blog/capo/2006/07/my-honest-dual-licensing.html>

[2] Driver, M. *Positions 2005: Open-Source Solutions Will Restructure the Software Industry*, Gartner Research, 2005 [www.gartner.com](http://www.gartner.com)

[3] Fitzgerald, B. *The Transformation of Open Source Software*, *MIS Quarterly*, Vol. 30, No. 3. December 2006,

<http://www.misq.org/archivist/vol/no30/issue3/Fitzgerald.html>

[4] Ganzaroli, A., Pilotti, L. *Natura e dinamica dell'open source (OS) tra creatività, ecologie del valore e fiducia*, Quaderni AIDA, Giuffrè editore, 2005

[5] Golden, B. *Why Open Source is Important to You*, NAVICA, TechnologyEvaluation.Com, April 2005

[http://www.technology-evaluation.com/Research/ResearchHighlights/FreeOpenSource/2005/04/news\\_analysis/NA\\_FS\\_XBG\\_04\\_13\\_05\\_1.asp](http://www.technology-evaluation.com/Research/ResearchHighlights/FreeOpenSource/2005/04/news_analysis/NA_FS_XBG_04_13_05_1.asp)

[6] Hillman, J. *Il Potere – come usarlo con intelligenza*, RCS Libri, 2002

- [7] Klinecwitz, K. *Innovativeness of open source software projects*, Tokyo Institute of Technology & Warsaw University Poland, 2005  
<http://opensource.mit.edu/papers/klinecwitz.pdf>
- [8] Koenig, J. *Seven open source business strategies for competitive advantage*, IT Manager's Journal, May 2004 <http://www.itmanagersjournal.com/feature/314>
- [9] Letellier, F. *Bringing Open Source Middleware to the Mainstream by Federating Business Ecosystems*, ObjectWeb Executive Committee INRIA, June 2005" [www.objectweb.org/wws/d\\_read/marketing/public/FLT-FederatingEcosystems-June05.pdf](http://www.objectweb.org/wws/d_read/marketing/public/FLT-FederatingEcosystems-June05.pdf)
- [10] Marchesi, M. *Il modello Open Source e come le PPAA possono trarne vantaggio*, Convegno PAAL 2007, Cagliari, marzo 2007  
[http://www.diee.unica.it/paal2007/AppendiceConvegnoOpenSource\\_CA\\_v2-2.pdf](http://www.diee.unica.it/paal2007/AppendiceConvegnoOpenSource_CA_v2-2.pdf)
- [11] Osterwalder, A., Pigneur, Y and Tucci, C. *Clarifying Business Models: origins, present, and future of the concept*, Communications of AIS, Vol. 15, May 2005  
<http://www.businessmodeldesign.com/publications/Preprint%20Clarifying%20Business%20Models%20Origins,%20Present,%20and%20Future%20of%20the%20Concept.pdf> (trad. autore)
- [12] Pilotti, L., Ganzaroli, A. *Proprietà Condivisa e Open Source*, FT Pearson Italia, 2007 (in fase di pubblicazione)
- [13] Pilotti L., Ganzaroli A. *Rileggere il marketing – strategie informative e gestione della conoscenza*, CEDAM, 2006
- [14] *Resoconto di dichiarazioni del ministro Luigi Nicolais*, Computerworld Italia Online, 17 gennaio 2007
- [15] Ruffatti, G., Cazzin, G. *Il futuro dell'Open Source a supporto del Business delle aziende e il Progetto SpagoBI*, quaderni di management n.21, maggio-giugno 2006  
<http://www.cwi.it/showPage.php?template=articoli&id=16686>
- [16] Thorn, S. *QA standards overview and Serono implementation choices and benefits*, EGEE'06 Conference, September 2006  
<http://indico.cern.ch/conferenceOtherViews.py?view=standard&confId=1504>
- [17] Valimaki, M. *Dual Licensing in Open Source Software Industry*, Systemes d'Information et Management 1/2003,  
<http://opensource.mit.edu/papers/valimaki.pdf>
- [18] Viele, O. *Software Libero, Opportunità o Minaccia per I system integrator?*, Homepage Engineering Ingegneria Informatica Anno VII n.8, Roma 2005.
- [19] Weiss, D. *Quantitative Analysis of Open Source Projects on SourceForge*, proceedings of the First International Conference on Open Source Systems Genova, 2005  
<http://oss2005.case.unibz.it/Papers/28.pdf>
- [20] Wheeler, D.A. *"Commercial" is not the opposite of Free-Libre/Open Source Software (FLOSS)*, December 2006 – January 2007,  
<http://www.dwheeler.com/essays/commercial-floss.html>