

Innovazione di impresa: realizzare software per la scuola

Stefano Penge

Lynx s.r.l.

Abstract

Questo articolo descrive la nascita in Italia, circa quindici anni fa, di piccole imprese dedicate allo sviluppo di software educativi di tipo innovativo. Cercheremo inoltre di capire la ragione per la quale nessuna di queste imprese, a oggi, continua a produrre software didattico. Infine proveremo a ipotizzare nuovi scenari possibili per la ricerca applicata al software educativo in Italia.

Parole chiave: ipertesti, software autore, imprese educative, politiche pubbliche.

Summary

In this article we describe the birth, in Italy fifteen years ago, of a number of small companies fully devoted to the development of innovative, educational software; we try to understand the reason why none of these companies is still developing educational software today. Finally, we try to draw the scenarios in which an applied research in this field would be possible in Italy.

Keywords: hypertexts, authoring software, educational companies, public policies.

Il contesto

Il rapporto tra tecnologie digitali e educazione in Italia negli ultimi dieci anni del millennio scorso ha subito un netto scatto in avanti. L'informatica non è più vista come una disciplina accanto alle altre; la multimedialità a basso costo comincia ad affacciarsi nelle scuole (e nelle case); riferimenti teorici come il costruttivismo vengono tenuti presenti anche quando si tratta di delineare delle applicazioni dei computer alla didattica. I saggi di Bolter e Landow sono tradotti e studiati, escono i primi scritti che «estendono» il concetto di testo (Scavetta, Corcione, Antinucci).

È in quel contesto che si apre la stagione delle piccole «tecno-imprese» educative: piccole realtà di soci-lavoratori legate allo sviluppo di pochissimi software tutti mirati al mercato educativo.

In alcuni casi significativi, è intorno a laboratori di ricerca universitari che su questi temi si coagulano dei piccoli gruppi di lavoro. Attratte dagli spazi teorici che si aprono intorno a testi come *Iperscuola* (Calvani, 1994) o il *Manuale di didattica multimediale* (Maragliano, 1994), che fissano l'uso in chiave educativa per i dieci anni seguenti di concetti come ipertesto e multimedia, e confortati da progetti pubblici di ricerca/sperimentazione a livello nazionale, come Multilab e il PSTD, che individuano nei laboratori universitari le punte più avanzate della ricerca, nascono una decina di piccole imprese che resteranno nel panorama italiano per i dieci/quindici anni seguenti. Citiamo almeno Lynx e Garamond a Roma, a Napoli Intermediammh, a Firenze IbisMultimedia, a Prato Knowledge Master, a Ivrea Incomedia.

Imprese di ricerca

Si tratta di imprese *sui generis*: sono create quasi senza capitali, da persone che spesso investono soprattutto il proprio lavoro. Sono costruite attorno a un'idea o intorno a un rapporto sociale, più che intorno a un vero *business plan*. Ne fanno parte insegnanti, grafici, programmatori, ricercatori, architetti, musicisti, uniti da passione e gusto per la novità. Se la forma societaria è quella della società a responsabilità limitata, la sostanza spesso rasenta quella cooperativistica o dell'impresa a conduzione familiare.

L'idea che sta dietro questi esperimenti (anche sociali) è quella di costituire delle realtà operative a metà tra il mercato e la ricerca pura. Se un laboratorio universitario può permettersi di muoversi in maniera orizzontale creando prototipi, un'impresa deve concretizzare in forma di prodotto quell'idea. Un prodotto — anche nel caso di un software — è un insieme di oggetti affidati a supporti diversi, di servizi che ne rendono possibile l'utilizzo, di pratiche connesse e condivise tra utenti. Un prodotto ha un ciclo di vita che non si conclude con la realizzazione della sua parte materiale ma continua nella promozione, nella distribuzione e nell'assistenza, come pure nella manutenzione, nella realizzazione di versioni successive, nel riposizionamento e nell'integrazione all'interno di prodotti successivi.

Tutte queste fasi richiedono risorse e professionalità che non competono a un centro di ricerca e spesso non sono presenti al suo interno. Di qui l'idea di prolungare il laboratorio in una struttura operativa esterna, autonoma, anche facendo ricorso a competenze esterne. D'altra parte, la caratteristica che unisce i soggetti di cui parliamo è il loro puntare su

prodotti innovativi, di cui non è ancora scontato il successo: si tratta di ambienti aperti, in cui l'utente/studente non è di fronte a serie di attività predeterminate, ma a contesti in cui può agire con un certo grado di libertà. Prodotti che non hanno senso al di fuori di una certa pedagogia e una certa pratica didattica innovative; la formazione dei formatori sembra quindi, insieme alla ricerca, l'altra gamba su cui queste società devono necessariamente poggiare. E su queste gambe, lentamente allontanarsi dai laboratori dove sono nate per cominciare una vita propria.

L'innovazione educativa come mercato

In un primo momento, la strada più semplice sembra essere quella della separazione tra creazione, realizzazione e produzione vera e propria. Le piccole imprese si pensano come laboratori di sviluppo di nuove idee in prodotti, e non ambiscono affatto a configurarsi come editori multimediali in proprio.

In questa fase nascono rapporti — o tentativi di rapporti — con editori grandi e piccoli: EDB, Anicia, Editori Riuniti, La Nuova Italia, Giunti, Petrini, Palombi... Si tratta di soggetti che non hanno all'interno competenze multimediali, ma che hanno un nome riconosciuto e soprattutto i canali della grande distribuzione, scolastici e non.

Alcuni prodotti riescono, grazie a questa alleanza, ad arrivare sul mercato anche in quantità massicce. Un esempio di questo piccolo miracolo è costituito da una serie di software dedicata alla scrittura digitale, intesa come ambito di apprendimento sperimentale e giocoso, che assume come nume tutelare Rodari. Il primo tentativo nasce come risposta alla richiesta di un editore di rilanciare sul mercato italiano, e in particolare nel canale delle edicole, alcune opere minori di Rodari, come il *Libro dei perché*; rilancio che avrebbe dovuto essere accompagnato da un prodotto multimediale, cioè da una videocassetta, sulla scorta di quanto stava succedendo in quel periodo. L'occasione è ghiotta: il team incaricato rilancia con la proposta di un floppy disk, supporto che è ormai sufficientemente diffuso, il cui contenuto non è un semplice racconto animato parallelo a quello scritto, ma un ambiente interattivo, nel quale i bambini possano giocare con la scrittura in tanti modi diversi, mettendo in pratica quello che soprattutto Maragliano e Vitali sostenevano da qualche anno.

Dal punto di vista della pedagogia, si trattava di un'interpretazione forte della multimedialità: non tanto o soltanto canali diversi aperti in contemporanea, ma parità di linguaggi (con una nuova attenzione allo spazio uditivo), rivalutazione delle competenze non verbali e delle logiche reticolari, «federazione dei media» che abbraccia pc e televisione, walkman e fumetto. Il canale dell'edicola permetteva di mantenere prezzi molto bassi, accessibili ai privati ma anche alle scuole; contemporaneamente, spingeva verso modelli seriali, sia in termini di dimensioni che di stile.

Ma la situazione globale — siamo nel 1996: l'Italia è alle prese coi parametri di Maastricht per entrare nell'euro — non è tanto rosea e gli editori cartacei neomultimediali non riescono a mantenere le promesse di un mercato ricettivo; le nostre piccole imprese di ricerca diventano giocoforza editori di se stessi: si inventano piani di marketing, canali distributivi, strategie promozionali. Partecipano a convegni e fiere (dal TED a Didamatica, dalla Fiera del libro per bambini di Bologna a Inforscuola), creano i loro siti web. Web che, in questi suoi primi anni di vita, viene visto soprattutto come un

canale di promozione e distribuzione di prodotti offline, più che come un prodotto esso stesso.

È in questi anni che diventa evidente il ruolo che potrebbe giocare un soggetto pubblico come il Ministero della Pubblica Istruzione in qualità di investitore. Il progetto Multilab (1996-1998) sembra far ben sperare: viene sviluppato o perfezionato ma soprattutto sperimentato grazie ad esso un certo numero di software didattici di più aziende italiane. Più avanti BDP/INDIRE si mostrerà attivo non soltanto come raccogliitore/valutatore di prodotti ed esperienze multimediali, ma anche come coordinatore/finanziatore di progetti di ricerca e sperimentazione didattica (come il Progetto Sperimentazione del 2002).

Uno degli errori, in prospettiva, da parte dei piccoli imprenditori educativi è stato probabilmente quello di non essersi saputi presentare in maniera collettiva come risorsa utile, forse unica, per lo sviluppo di una cultura didattica legata alle tecnologie; risorsa che, forse, meritava di essere mantenuta in vita attraverso un accordo quadro che tenesse conto delle specificità di ciascuno, al posto di una serie di contratti individuali che non potevano garantire il mantenimento di linee di ricerca, di piani di sviluppo a medio termine, di risorse dedicate all'assistenza, alla manutenzione e alla formazione.

Gli strumenti

La velocità di evoluzione delle tecnologie digitali è tale che un minimo di contestualizzazione è necessaria. Nel 1994 è uscito Netscape e l'anno dopo Internet Explorer; Frontpage è del 1996, nello stesso anno in cui in Italia parte il progetto Multilab. Per DreamWeaver bisogna aspettare l'anno seguente. Solo nel 1999 vengono pubblicate le specifiche dell'HTML 4, che è sostanzialmente quello in uso oggi. Google è appena nato e per Wikipedia bisognerà aspettare ancora fino al 2001. Windows è praticamente l'unico sistema operativo in circolazione, e su Windows lo strumento più usato per fare ipertesti è probabilmente Asymetrix ToolBook, anche perché i fondatori di Asymetrix e Microsoft sono amici e il runtime di Toolbook 1.0 è regalato insieme a Windows 3.0.

ToolBook è un clone di HyperCard per Windows. Ha però delle caratteristiche interessanti che lo rendono lo strumento ideale per i prodotti ipertestuali che sono caratteristici di questo periodo. E forse capirne le potenzialità e i limiti è un passo essenziale per cogliere il fenomeno di cui ci stiamo occupando.

La sua metafora di fondo è molto forte e piuttosto invasiva: si tratta del «libro». In TB ogni applicazione è vista come una piramide composta da uno o più «libri», ognuno dei quali è diviso in capitoli, ogni capitolo composto da pagine. Se immaginiamo questa struttura come una sovrapposizione di livelli trasparenti, dal basso verso l'alto, abbiamo anche la gerarchia degli oggetti: un oggetto, una proprietà o un metodo appartenente al livello più basso è visibile in tutti i livelli superiori. Mentre negli strumenti classici le interfacce dei moduli di un'applicazione sono *maschere*, cioè pagine con buchi, attraverso i quali vengono caricati e visualizzati i dati sottostanti, in TB al contrario i dati individuali (testi, immagini, oggetti grafici e sonori) sono disposti *sopra* le strutture comuni. Questa architettura permette di creare applicazioni anche complesse semplicemente disegnando e disponendo oggetti, secondo un ordine che si presenta piuttosto naturale da usare in un contesto scolastico.

La metafora del libro (o dovremmo dire del quaderno?), quasi completamente autoconsistente, permette di usare ToolBook senza essere degli informatici, senza cioè avere idea di come vengano memorizzati i dati, di come venga gestita la memoria, ecc. Si crea uno spazio comune tra insegnanti, studenti e professionisti che si trovano a condividere uno strumento, una metafora, un linguaggio, anche se ognuno può usarlo e approfondirlo a livelli diversi. Lo stesso ambiente di sviluppo può essere usato per il più semplice ipertesto scolastico come per sofisticate applicazioni che gestiscono dati multimediali georeferenziati. Non è un caso che proprio su TB venga organizzato a Panicale nel 1999 un seminario residenziale di autoformazione in cui a turno tutti i partecipanti hanno la possibilità di essere docenti e discenti.

Un altro aspetto importante di TB è che possiede un linguaggio interpretato; è possibile, durante l'esecuzione di un programma, creare oggetti, assegnare proprietà ma anche scrivere nuovi metodi. Di qui la facilità con cui è possibile creare applicazioni che evolvono, il cui comportamento è fundamentalmente deciso dall'utente. Di qui ancora la possibilità di creare una classe speciale di applicazioni: i software *autore* educativi, il cui contenuto non è predeterminato dal programmatore, ma deve essere deciso e realizzato dall'utente. Solo la metafora, in alcuni casi, viene fissata, come indirizzo di massima e come limitazione delle funzioni presenti.

È interessante seguire l'impatto sulla cultura educativa italiana di quegli anni di una caratteristica così squisitamente tecnica. TB era uno strumento relativamente caro, almeno per il mercato scolastico italiano, ma aveva la particolarità di avere un *runtime* (l'interprete, la parte necessaria per far funzionare le applicazioni) ridistribuibile gratuitamente. Per le ragioni dette sopra, con il solo interprete e poche righe di codice era possibile realizzare applicazioni che, a partire da uno scheletro, erano in grado di crescere. In pratica, era possibile creare dei mini-toolbook che potevano essere venduti a un prezzo molto inferiore all'originale ma ne avevano tutte le potenzialità principali, almeno quelle utilizzabili in una scuola. Cavallo di battaglia di alcune, se non tutte, le piccole imprese di cui parliamo, sono proprio una o più applicazioni di questo tipo, che permettevano di creare mappe, vocabolari, atlanti, libri multimediali ovvero ipertesti.

Sembrava l'uovo di Colombo, perché promuoveva il prodotto completo (TB) attraverso dei fratelli minori a basso costo; e invece fu all'origine di una polemica tra la casa madre, la Asymetrix, e una delle imprese, cui fu intimato di smettere di distribuire la prima versione di uno dei software autore più conosciuti in Italia a causa di una presunta violazione del contratto.

Quello che qui ci interessa sottolineare è che anni prima della diffusione in Italia del paradigma open source si cominciava a sentire l'esigenza di colmare lo spazio tra informatica «ufficiale» e informatica «artigianale», tra programmatori professionisti e ricercatori in campo educativo, docenti, genitori e studenti. Anni prima che il web 2.0 inventasse (o riscoprisse) il ruolo dell'utente-autore, almeno nel campo educativo questo utente non passivo ma creativo, poco professionale ma impegnato, esisteva ed era ben presente come target. Una volta compreso che il motore educativo dell'ipertestualità non era la fruizione, ma la creazione di link e nodi, la disponibilità di strumenti economici in grado di soddisfare questa nuova richiesta era una condizione essenziale per lo sviluppo. In un'era in cui l'HTML cominciava appena ad affacciarsi come standard, ma meno del 50% delle scuole italiane aveva un sito web, l'uso di un software autore ipertestuale diventava un passaggio obbligato.

Curiosamente, l'onda dell'opensource e del software libero non ha facilitato la vita di queste imprese. Nel momento in cui diventavano disponibili software didattici «free» (senza che in un primo momento fosse chiara la differenza tra libero, gratis e aperto), italiani ma soprattutto stranieri, e mentre internet rendeva finalmente accessibile tutto quello che veniva sviluppato da insegnanti di buona volontà, diventava sempre più difficile *vendere* del software didattico. Nel nome della libertà della cultura, o sulla base di una analisi della scarsità delle risorse delle scuole, gli stessi insegnanti cominciarono a guardare con sospetto le software house che pretendevano di farsi pagare il software didattico dalle scuole. Perché acquistare quello che si poteva scaricare?

Si tratta di uno snodo cruciale. Purtroppo in quel momento non viene fatta un'analisi delle differenze (positive e negative) tra il software artigianale e quello commerciale, né ci si domanda quali saranno le prospettive se l'unica risorsa dedicata alla manutenzione e sviluppo del software didattico è la buona volontà.

A quel punto due strade si presentavano: quella di rilasciare il software didattico sotto una licenza GPL e renderlo scaricabile liberamente, cercando però altri canali di finanziamento; oppure battere la strada del «CD allegato» ai testi scolastici, il cui costo viene assorbito dal cartaceo, ma assumendo come limite quello del software ancillare, l'eserciziario con autocorrezione, abbassando i costi di produzione attraverso la localizzazione di software sviluppati in altre lingue per altri Paesi.

In entrambi i casi, il progetto, lo sviluppo, la distribuzione e l'assistenza relativa al software didattico in Italia si fermano.

Conclusioni?

Siamo alla fine della nostra storia: le piccole imprese sono scomparse o, in ogni caso, non progettano e sviluppano più software didattico educativo innovativo. Poco ci importa del loro destino: ma che ne è del software educativo stesso?

Possiamo rinunciare per sempre all'idea di lavorare, a scuola, a casa, e in ogni contesto in cui si possa usare un computer, su ambienti software specificamente dedicati a questo scopo? Possiamo considerare la professione di progettista di software didattico un mestiere morto, che non vale la pena di insegnare/apprendere?

Chi scrive, avendo dedicato la maggior parte della propria vita lavorativa a riflettere, insegnare e scrivere codice per questo obiettivo, è ovviamente di parere contrario.

In realtà, con l'estinzione di queste forme di vita poco adatte, non si ferma del tutto lo sviluppo di software didattico, anche se da parte di soggetti diversi: insegnanti che a livello individuale, sulla base della propria esperienza e usando il proprio tempo libero, scrivono software e lo distribuiscono sul proprio sito web. Indipendentemente dalla qualità dei software stessi, che naturalmente può essere notevole, è chiaro che non si tratta più di prodotti, nel senso definito sopra: gli autori non possono garantire assistenza, documentazione o formazione.

D'altra parte, questa piccola storia ha forse dimostrato che i soli finanziamenti pubblici non sono sufficienti se non fanno parte di un progetto complessivo che comprende la formazione, coinvolge categorie diversi di attori, stabilisce linee guida per la valutazione lungo l'arco della vita del software. La via dell'opensource sembra l'unica percorribile, a

patto che non se ne faccia solo una bandiera. Il modello opensource funziona non solo sulla base della gratuità dell'impegno. Il tempo dedicato da analisti, progettisti, programmatori, tester se non è pagato immediatamente e direttamente dagli utenti come licenza, viene però almeno giustificato dal riconoscimento dell'esperienza acquisita sia nell'universo chiuso dei programmatori, sia sul mercato. È possibile ridirezionare i flussi economici dall'acquisto delle licenze alla richiesta di servizi (di assistenza, di personalizzazione, di formazione); ma non è possibile postulare l'esistenza di software educativo senza domandarsi come verranno retribuiti i lavoratori che lo producono.

Bibliografia

- Badii V., Leonetti F. e Rotta M. (1995), *Costruire ipertesti. Guida all'uso didattico di ToolBook*, Roma, Garamond.
- Calvani A. (1990), *Dal libro stampato al libro multimediale*, Firenze, La Nuova Italia.
- Calvani A. (1994), *Iperscuola. Tecnologie e futuro dell'educazione*, Padova, Muzio.
- Maragliano R. e Vitali L. (1989), *Videoscrivere in classe*, Roma, Editori Riuniti.
- Maragliano R. (1994), *Manuale di didattica multimediale*, Bari, Laterza.
- Maragliano R., Martini O. e Penge S. (1994), *I media e la formazione*, Roma, La Nuova Italia Scientifica.
- Penge S. (1993), *Io Bambino Tu Computer*, Roma, Anicia.
- Penge S. (1996), *Storia di un Iper testo. Pensare e scrivere in forma di rete*, Firenze, La Nuova Italia.