

# Il respawn di Infomedia ( $\text{\LaTeX}$ -based)

*Emmanuele Somma*

## Sommario

Infomedia, noto editore italiano di riviste di programmazione, rinasce anche grazie al software libero. Per comporre le sue riviste si affida a  $\text{\LaTeX}$  e ad altri strumenti discussi nell'articolo.

## Abstract

Infomedia, famous Italian publisher of programming magazines, has born again thanks to free software. To typeset its magazines it relies on  $\text{\LaTeX}$  and some other tools discussed in the paper.

## 1 Infomedia e l'editoria per l'informatica

Per un breve periodo della mia vita, tra la laurea e il primo concorso come programmatore al Ministero della Giustizia, ho lavorato in una piccola azienda, con sede a Ponsacco vicino Pisa.

Era un'azienda un po' strana: piccola, pochi dipendenti, giovane. Il più anziano poteva avere più o meno la mia età, circa 25 anni. E il cui fondatore era più giovane di me.

Pubblicavano riviste. Le uniche riviste italiane di programmazione, al tempo, nate quando i magazine del settore decisero di mollare gli spazi in cui pubblicavano listati e programmi per buttarsi nel più ricco mercato dell'informatica aziendale, prove hardware, marchette commerciali e altra paccottiglia del genere. Mainstream ma noioso, a vederla con gli occhi di un programmatore.

Si può certamente dire che intere generazioni di professionisti italiani trovavano in Computer Programming, DEV, Login e Visual Basic Journal (VBJ) un'occasione privilegiata per imparare veramente cosa fosse un computer, conoscere come usarlo e mantenersi aggiornati.

Credo di averci lavorato non più di sei mesi. Con un piede dentro il processo editoriale e uno dentro il nascente settore dei servizi Internet. Quella Internet fatta ancora di modem, provider in teleselezione, dove non era raro trovarci ancora Gopher e Veronica.

In particolare mi occupai di un progetto, dopo la mia partenza (e di quella del fondatore, quasi contemporanee), mai più realizzato: la realizzazione di una 'community' degli sviluppatori italiani. Al tempo quasi non esisteva nemmeno la parola community!

Sei mesi che hanno cambiato la mia vita.

Ho continuato a collaborare con loro anche dopo il mio trasferimento a L'Aquila, e poi a Roma. Per un certo tempo mi occupai dell'impaginazione delle riviste online. Nel frattempo tante cose erano cambiate nel mondo informatico: erano cambiate per loro, erano cambiate per me. Rimase una grande amicizia ma poi, quando mi è capitato di proporre una nuova iniziativa editoriale su Linux, ho dovuto cercare nella concorrenza un terreno più fertile per le mie idee. È però rimasta l'amicizia di un tempo.

Per qualche anno ho portato avanti la mia rivista con l'altro editore, tra alti e bassi, talvolta discutendone più con gli amici di Ponsacco, sicuramente più focalizzati sulle prospettive professionali, che non con il mio editore, con una visione più generalista.

Poi un nuovo concorso, maggiori impegni, qualche dissapore con l'editore e... sì, è bello sapere che le iniziative che contribuisce a realizzare, in un certo modo, sopravvivono anche senza di te. Che non hai perduto vecchi amici, ma te ne sei fatti di nuovi.

Ancora una volta il mondo informatico però era cambiato, ed era cambiato molto il modo di fare editoria tecnica. E io uscivo da un'iniziativa con un po' di esperienza in più in questo campo ma sinceramente pensavo di aver chiuso un capitolo. E per celebrarne questa soddisfacente chiusura spedii un lungo messaggio ai miei 'pari', cioè alcuni editor-in-chief di riviste italiane, che avevo avuto il piacere di conoscere nel tempo e che, come me, avevano vissuto relazioni non sempre edificanti con i loro editori.

Senza scendere nei dettagli il messaggio trattava della qualità delle relazioni interne alle redazioni, alla riduzione dei budget per le collaborazioni esterne e quindi alla nascita, mai vista prima, dei redattori interni 'generalisti' buoni a scrivere articoli su tutto e per tutto: i riempitivi. Lamentavo della nascita delle riviste-riempitivo per colpa della riduzione dei budget. Agli editor-in-chief miei amici scrissi invece dell'esperienza molto soddisfacente, e assolutamente anticipata dovuta al tema proprio della mia rivista, di ottenere articoli 'liberi', basati cioè su licenze di distribuzione 'open source' dalla 'Comunità'. Io stesso, fin dal 1998, avevo scritto in GNU FDL, dopo lunga discussione con l'editore, e poi compiuto una transizione alle neonate e poco diffuse Creative Commons.

Un passaggio di questa lettera trattava anche del rapporto con le fasi produttive, impaginazione e stampa, che da sempre è stata una delle croci che gli editor hanno dovuto accollarsi e che in generale

contribuiva a far decadere la qualità dell'intero processo editoriale per l'assoluta rigidità, e talvolta incompatibilità con le proprie stesse norme, degli impaginatori professionisti capaci di combinare qualsiasi disastro in articoli altrimenti perfetti. Sostenevo si potesse fare una rivista senza ricorrere agli impaginatori.

Nel frattempo l'anima commerciale delle riviste cresceva a causa della riduzione delle disponibilità aziendali, per la necessità di mantenere alti gli utili d'impresa e per il malinteso confronto con Internet che portava gli editori a rafforzare, invece che rilassare, le misure restrittive sul copyright. I contratti di edizione, di cui feci incetta tra gli amici in quel periodo, diventavano sempre più assurdi. Era veramente difficile firmarli a cuor leggero.

Si aggiunga a ciò quell'oscuro periodo in cui, nella generale riduzione degli investimenti pubblicitari sulla carta stampata, spiccavano in positivo le elargizioni di un'unica azienda, inutile far nomi, che contribuirono a drogare, e non poco, anche il contenuto editoriale.

In buona sostanza, era facile la previsione di una crescente disaffezione dei lettori per le riviste a contenuto tecnico. Non però per l'incombenza di Internet, quanto per il vero tracollo qualitativo dei contenuti. Solo una transizione verso altri modelli di business, probabilmente con minore componente commerciale, avrebbe potuto permettere il 'salvataggio' di una presenza editoriale in campo professionale, mutuando per quanto possibile le esperienze internazionali delle associazioni professionali del settore. Gli editori però avrebbero preferito, stanno preferendo, morire asfissati che iniziare questi processi.

Quando, dopo qualche anno, i miei vecchi amici di Ponsacco tirarono fuori dal cilindro tutte quelle considerazioni per verificare quanto fosse possibile riorganizzarsi in base a quelle considerazioni ormai era troppo tardi.

Il Gruppo Editoriale aveva già cessato la pubblicazione di tre delle cinque testate, l'emorragia di dipendenti e collaboratori era già molto avanzata e non si era chiaramente più in grado di portare avanti la produzione delle altre riviste.

Negli ultimi 15 anni Computer Programming, DEV, VBJ e Login avevano contribuito a far conoscere e definire il mondo della programmazione in Italia rendendolo popolare anche nelle scuole e tra i ragazzi. Alla loro chiusura il panorama italiano dell'editoria professionale nel campo della programmazione era diventato desolante.

L'editoria tradizionale sfoggiava, sul tema programmazione, praticamente un unico concorrente con una natura fin troppo commerciale, qualche altra rivista sfociava sul sistemistico, specie nel campo Linux.

In Internet invece erano proliferate una grande quantità di iniziative lodevoli (siti, blog, aggregato-

ri), alcune di buon livello qualitativo, nessuna delle quali però sembrava in grado di diventare un vero punto di riferimento per una comunità nazionale che, d'altro canto, non pareva avere una grande 'coscienza di ruolo', se non di classe. Inoltre duravano il tempo dell'autopromozione del coordinatore che, passato a impegni più gravosi, non riusciva a mantenere continuità alla produzione o a passarla ad altri.

Una rivista periodica sembra poca cosa: d'altro canto oggi Internet non ci dà accesso a risorse infinite e tutte alla portata di mano?

Quello che c'è alle spalle di una rivista periodica è un lavoro che andrebbe molto meglio considerato. Poi capita che una volta conosciuto non si può non apprezzare.

Gli obblighi imposti dalla periodicità fanno mantenere attiva una macchina di selezione ed adattamento dell'informazione che produce con continuità un contenuto 'abbastanza buono' (good enough).

Se i processi di selezione ed adattamento sono validi, non è difficile accorgersi del valore, anche economico, di quei contenuti perché, ad esempio, permettono ad alcuni lettori di risparmiare una gran quantità di tempo e risorse per 'tenersi in pari', non tanto nei campi in cui sono già competenti (che copriranno sempre meglio di quanto potrà fare, periodicamente, una rivista) ma tutti gli altri vicini o afferenti su cui da un lato non possono permettersi di rimanere ignoranti, dall'altro non possono sprecare una grande quantità di tempo anche solo per cercarsela, l'informazione.

Una rivista equilibrata e aggiornata permette inoltre sempre di tenere sott'occhio una possibile 'next big thing' senza la necessità di ingurgitarsi la lettura di migliaia potenziali fonti d'informazione, e su cui ogni volta fare la tara.

I contenuti 'good-enough', poi, hanno un incredibile valore per chi vuole avvicinarsi ad un mondo e diventano risorse preziose in un'ottica d'istruzione e di mentoring. Istruzione e mentoring che dovrebbero essere l'obiettivo proprio di una comunità professionale *seria*.

Non mi spingo oltre in queste considerazioni; sarò felice di discuterne eventualmente nell'ed-board di Infomedia.

La perdita di uno spazio di selezione e adattamento, come erano state le riviste Infomedia, è stata quindi significativa.

Erano venuti così meno gli unici luoghi italiani che negli anni avevano mostrato indipendenza e professionalità nel 'rappresentare' questo ambiente in modo aperto e popolare, senza dover ricorrere ai gradi e ai galloni accademici che non sempre, in Italia, hanno selezionato la qualità, ma soprattutto è venuta meno una costante attenzione nel trattare il non semplice tema della programmazione e della sistemistica senza essere trascinati troppo da


 Search this site

Sei capitato sul sito del Turing Club, la prima (e unica) [associazione italiana](#) dei programmatori professionisti, e appassionati di programmazione in Italia.

Ci ispiriamo ad [Alan Mathison Turing](#), perché tra i padri della computer science è stato il primo a capire che la nascita del calcolatore digitale avrebbe fatto sorgere una nuova leva di professionisti e che si sarebbe sviluppata una loro cultura e professionalità specifica. Turing realizzò il primo linguaggio e fu quindi senza dubbio il primo programmatore in senso moderno.

Poiché troviamo che la figura di Turing, in Italia, sia ricordata solo legata alla vasta aneddotica sulla sua vita turbolenta, o un po' per i suoi importanti contributi scientifici, ma troppo poco rispetto al suo ruolo di innovatore delle strutture sociali e culturali, noi programmatori e sistemisti vogliamo segnare, con questa attribuzione, il ringraziamento imperituro a questa fondamentale figura anche della nostra cultura, e per la quotidianità della nostra vita.

Il Turing Club tiene alla promozione della cultura informatica, specie tra le giovani generazioni e per questo si impegna. L'informatica ha infatti cambiato la vita della nostra Società, ma troppo spesso si immagina che alle spalle di questa innovazione ci sia una sorta di incomprensibile magia. Questa "magia" non è che il frutto di giorni e notti di programmazione appassionata il cui valore sociale è ben superiore al guadagno che se ne ricava.

In una società come quella italiana poi, per ragioni anagrafiche e generazionali, per l'assenza di un sistema di credito avanzato e per la miopia della classe direttiva e politica, il talento, la competenza e l'innovatività dei programmatori piuttosto che essere premiata, viene giorno dopo giorno disconosciuta e marginalizzata, con il pratico risultato di spingere l'Italia nel Medioevo tecnologico.

Chinque tu sia, quali che siano i tuoi strumenti di lavoro, il tuo linguaggio di programmazione, la tua piattaforma o framework, il tuo sito o il tuo programma... [questo](#) è il Turing Club



FIGURA 1: Messaggio "autopromozionale" del Turing Club

logiche commerciali legate ad altre attività business dell'editore. Non c'era più nessuno "sul pezzo" della programmazione.

Il Gruppo Editoriale Infomedia era riuscito fino all'ultimo a rimanere un 'editore puro', completamente focalizzato sul tema. E se da un lato gli si poteva imputare una certa inerzia nel prendere coscienza delle mutate condizioni del contesto e una certa inazione nei confronti del fenomeno Internet, dall'altra non si poteva non riconoscergli che era rimasta comunque fedele al proprio pubblico, mentre gli altri correvano dietro alla completa de-specializzazione, per rincorrere più alte vendite e sopportare le bizzarrie di un sistema di distribuzione in edicola che è un vero sconcio italiano. Ma soprattutto aveva negli anni prodotto una mole incredibile di informazione, dando voce ad oltre 800 autori, e realizzando molte migliaia di articoli tecnici, presentazioni di prodotti, tecnologie, protocolli, strumenti di lavoro, tecniche di sviluppo e semplici trucchi e trabocchetti, con oltre 6 milioni di copie distribuite e trentamila pagine stampate avevano fatto di questa impresa la più grande ed influente realtà dell'editoria specializzata nel campo della programmazione e della sistemistica.

Lo stesso rapporto con l'advertising, argomento a cui io personalmente ero molto sensibile essendo in effetti stato causa del raffreddamento dei rapporti con il mio precedente editore, era rimasto di gran lunga più sano che non in altri casi.

D'altronde l'impossibilità di mantenere le proprie attività era avvenuta non già per una riduzione degli abbonati, per la maggior parte clienti istituzionali, università e aziende, grandi e piccole, rimasti sostanzialmente costanti nei precedenti dieci anni, ma per i soliti problemi di liquidità in cui vengono a trovarsi piccole aziende del genere

strozzate dal sistema di credito, dovuti peraltro ad attività collaterali alla produzione delle riviste vere e proprie.

Alla perdita di questa realtà molti programmatori non hanno saputo e voluto adattarsi.

Già dai primi momenti, anche grazie alla disponibilità dei soci del precedente Gruppo Editoriale, che pure apparivano poco interessati a continuare con l'impresa, si è ragionato sul come permettere alla Comunità dei programmatori di non disperdere questo patrimonio. Promotore di questo tentativo un gruppo di programmatori e sistemisti, che in modo informale e talvolta conviviale e un po' goliardico, aveva al tempo già fatto proprie considerazioni circa l'importanza di rendere più popolare il 'pensiero informatico' nella società italiana: il Turing Club. Nella figura 1 è riportata la pagina web del manifesto del gruppo.

Non era però ragionevole pensare ad infondere nuova linfa nella vecchia azienda costruita su un modello alquanto tradizionale di editoria.

Il tentativo di una nuova Infomedia è stato quindi non tanto quello di trovare il modo di "continuare" tal quale, quanto piuttosto di "innovare" anche i metodi editoriali stessi con lo stesso talento che solitamente i programmatori infondono nelle loro iniziative.

## 2 Una nuova Infomedia

Molti pensano che inesorabilmente le riviste cartacee e addirittura il modello di produzione delle riviste periodiche sia tramontato per sempre di fronte all'innegabile successo di Internet.

Chi si è ritrovato attorno alla nuova Infomedia non la pensa così e ritiene che non solo esista ancora uno spazio, ma che la nuova Infomedia rappresenti un'occasione unica per mettere in atto modelli di

produzione mutuati dalle migliori esperienze della Rete.

La nuova Infomedia si è quindi organizzata attorno ad una idea di Impresa Sociale di Comunità, partecipata da programmatori e sistemisti, separando le attività di gestione dell'informazione realizzate da un board comunitario professionale e quelle di produzione gestite da una impresa strumentale (Infomedia Editori Coop) anch'essa non a fini di lucro.

Dall'inizio di quest'anno si è riuscito a riprendere le pubblicazioni di:

- CP Computer Programming (bimestrale)
- DEV developing software (bimestrale)

Coloro che si sono lanciati in quest'impresa, come volontari e per puro spirito di servizio, hanno ancora molto da imparare e i risultati sono certamente migliorabili, ma senza dubbio è già possibile vedere quanto sia importante il lavoro svolto fin qui.

È opinione comune dei lettori che hanno continuato a leggere le riviste che il livello qualitativo medio dei numeri fin qui prodotti sia cresciuto notevolmente. Per non parlare delle collaborazioni internazionali che abbiamo avuto il piacere di ricevere: Peter Norvig, Paul Graham, Richard P. Gabriel, Neal Gafter, Walter Bright, Joel Spolsky.

Tanto per citare le collaborazioni degli ultimi sei mesi.

Va notato il coinvolgimento che si sta iniziando a creare attorno all'iniziativa. Dal primo numero, gestito da un risicato manipolo e tre mesi di lavoro per la pubblicazione, agli ultimi, che stanno mantenendo le scadenze mensili alternate di due riviste. Tutto grazie alla contribuzione di una comunità composta da quasi cinquanta persone come editor, autori, revisori, sviluppatori, ma soprattutto come membri di un 'editorial-board' aperto e pubblico, che in un certo senso sta iniziando a guidare l'evoluzione dell'iniziativa (la figura 2 mostra il sito della nuova Infomedia).

### 3 La sfida

La sfida più grande di questo 'sistema' è rappresentata dal modello di open-publishing che si sta realizzando. Un modello che riesca a coniugare la contribuzione libera e aperta, secondo uno schema mutuato dalle migliori esperienze delle riviste scientifico-professionali, specie nel mondo anglosassone, e il livello qualitativo e le garanzie di continuità delle riviste commerciali. Con la prospettiva realistica di ritornare ad una distribuzione cartacea e con la velleità di superare, anche, la

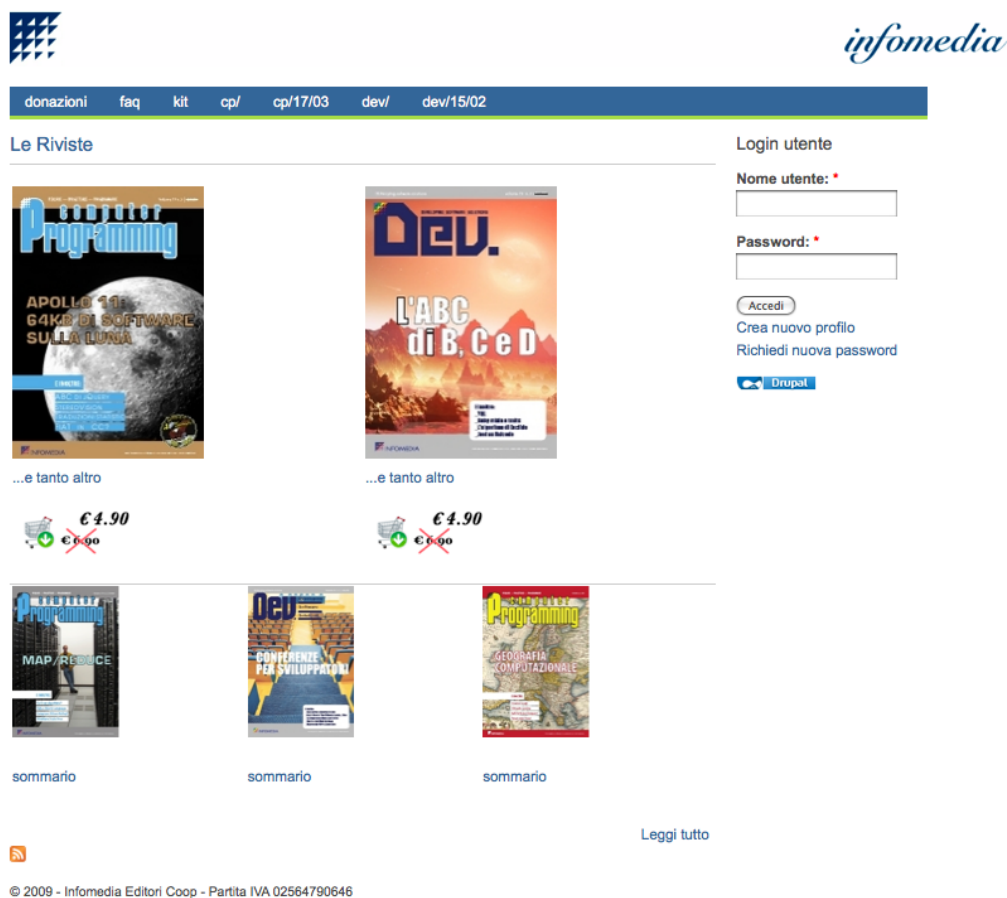


FIGURA 2: Home page del sito della nuova Infomedia

diffusione precedente, proprio grazie all'interazione con la comunità.

A ciò andrebbe aggiunta una seria riflessione circa la prospettiva della distribuzione dei contenuti in una modalità completamente open-access, attraverso licenze Creative Commons.

Ciò che però rende esclusiva questa esperienza, a parte le prospettive numeriche dei sottoscrittori e la diffusione nella pur piccola realtà italiana, sono gli strumenti adottati per realizzare la promessa di open-publishing, e quindi abbattere le barriere all'ingresso del lavoro editoriale e semplificare notevolmente sia la contribuzione di articoli sia il percorso di generazione del 'prodotto finale', ovvero il PDF pronto per la stampa cartacea.

In questo il sistema L<sup>A</sup>T<sub>E</sub>X gioca un ruolo molto rilevante. E vista la sua centralità gli deve essere tributato un plauso speciale proprio come tecnologia abilitante.

In effetti la produzione della nuova Infomedia non fa che seguire l'esperimento condotto dalla testata americana Free Software Magazine (FSM), costruita con una class L<sup>A</sup>T<sub>E</sub>X realizzata specializzando le classi `journal` e `paper` da Gianluca Pignalberi. L'articolo di Gianluca (PIGNALBERI, 2005b) spiega fin troppo bene i problemi a cui va incontro chi volesse impaginare una rivista con L<sup>A</sup>T<sub>E</sub>X.

Va subito detto che, pur nella sua veste internazionale, anche l'iniziativa FSM è in un certo senso figlia della 'vecchia' Infomedia. Infatti il direttore, Tony Mobily, è stato per un lunghissimo periodo collaboratore, socio e responsabile di alcune testate Infomedia. Oltre che, per me, amico fraterno.

Se si può parlare di una 'scuola Infomedia', sia io che Tony, e lui molto più di me, ci siamo sicuramente diplomati lì.

La nuova Infomedia deve quindi moltissimo a Tony Mobily e Gianluca Pignalberi: non solo tutta quella familiarità e amicizia che una comune passione può generare, ma proprio l'aver messo a nostra disposizione tutto il codice necessario e l'aiuto per comprenderlo ed imparare ad usarlo.

Sebbene nel frattempo la stessa FSM abbia abbandonato la produzione cartacea e anche l'impaginazione L<sup>A</sup>T<sub>E</sub>X, la nuova Infomedia ha invece raccolto il testimone e lo ha integrato in un sistema più completo, per affrontare il problema della produzione distribuita della rivista.

Sono tre le differenze principali introdotte nel sistema Infomedia (chiamato dagli editor 'making' poiché la directory principale di lavoro è così denominata):

- l'uso di un file di input in formato solo testo con markup semplificati (tipo Wiki o Textile) al posto del L<sup>A</sup>T<sub>E</sub>X stesso chiamato file `.tchl.txt`, e quindi di uno strumento di traduzione da `.tchl.txt` ad un apposito sottoinsieme di L<sup>A</sup>T<sub>E</sub>X realizzato in Emacs Lisp;

- l'adozione di un programma d'interazione command-line per l'esecuzione di tutti i comandi necessari alla composizione e, in parte, al publishing della rivista, realizzato in Python;
- l'uso di SVN, non solo nello sviluppo dei sorgenti del software realizzato e delle macro L<sup>A</sup>T<sub>E</sub>X, ma per la gestione complessiva del flusso editoriale, articoli, figure, elementi grafici e file di configurazione.

Il sistema realizzato ha ormai raggiunto un buon livello di maturità, sono stati realizzati già sei numeri delle riviste, per un totale di più di 350 pagine lavorate completamente con questo sistema.

#### 4 I markup nel file `.tchl.txt`

Uno degli ambiti di miglioramento del sistema proposto da Free Software Magazine era senza dubbio il loro modello di redazione dei singoli articoli.

Per contribuire a FSM bisognava in definitiva costruire un file L<sup>A</sup>T<sub>E</sub>X, contenente un sottoinsieme di comandi molto limitato.

Tutti questi singoli file venivano poi collezionati in una sottodirectory comune e poi richiamati dal file principale con delle macro d'inclusione.

La composizione finale della rivista prevedeva quindi la realizzazione di un documento principale L<sup>A</sup>T<sub>E</sub>X, di classe `fsmjournal`, in cui fare l'input degli articoli ad uno ad uno attraverso specifiche macro. Era responsabilità dell'editor costruire tabelle file introducendovi gli articoli e tutti gli altri componenti necessari (editoriali, sommario, pagine pubblicitarie, ecc. ecc.).

Questo lavoro, quindi, non poteva che essere delegato ad un T<sub>E</sub>Xnico accorto, che intervenisse sugli articoli, sul file principale e talvolta sulle macro per risolvere 'creativamente' gli inevitabili problemi di impaginazione.

L'innovazione introdotta da TCHL è stato l'utilizzo di un formato testuale molto semplice per la redazione degli articoli. Non un L<sup>A</sup>T<sub>E</sub>X ridotto quanto piuttosto un formato testuale wiki agli stereoidi.

Il progetto BHL (<http://www.nongnu.org/bhl/>) di Bastien Guerry, già dal 2002 mirava a fornire all'utente Emacs un semplice sistema di redazione di documenti basato su un markup implicito e minimale, simile a quello usato nella redazione di un Wiki, che poi poteva essere automaticamente trasposto in formato HTML, SGML (Linuxdoc), Texinfo o L<sup>A</sup>T<sub>E</sub>X.

Avevo utilizzato una forma leggermente adattata di BHL per l'editing di Linux Magazine, accordandomi con il settore impaginazione a cui fornivo gli articoli in formato con un markup HTML minimale per il corpo dell'articolo, e un file testuale a parte contenente tutte le informazioni identificative dell'articolo (titolo, sottotitolo, autore, didasca-

lie, ecc.) da copia-incollare linea per linea nelle gabbie d'impaginazione. Una pazzia che solo un Cercopithecus-Impaginator poteva immaginare.

Il passaggio da questo alla compilazione diretta del file L<sup>A</sup>T<sub>E</sub>X preteso da fsmjournal fu fatto, ovviamente, in uno snap.

L'idea era quella di riuscire a 'nascondere' alla vista degli editor la presenza del L<sup>A</sup>T<sub>E</sub>X, non tanto per evitar loro di dover acquisire competenze in questo sistema, quanto per impedirgli di pasticciare con ambienti e comandi poco consoni alla composizione, vanificando il lavoro anche degli altri editor, potenzialmente al lavoro sulla stessa rivista in contemporanea.

Il vantaggio principale stava quindi nel cablare nel codice Lisp, una volta per tutte, le traduzioni testo→L<sup>A</sup>T<sub>E</sub>X, in particolar modo in relazione a quegli ambienti classici di una rivista del genere (figure, listati, box).

Una volta tradotto il markup file `.tchl.txt` in uno `.tex`, era poi possibile attraverso la linea di comando richiedere la composizione dell'articolo o, calato l'articolo in una struttura di rivista, ricompilare l'intera rivista.

Non c'è bisogno di scendere nei dettagli del markup perché, in realtà, quello adottato poco aggiunge a sistemi simili. Essendo stato realizzato prima di tanti altri è, ovviamente, incompatibile e pure un po' specializzato. Allo stato attuale lo consideriamo già un po' legacy e ci piacerebbe poterlo uniformare a qualcosa di più comune, in modo anche da poter riusare strumenti di traduzione testo→L<sup>A</sup>T<sub>E</sub>X differenti (purtroppo il Lisp gode di poco seguito nella community, per ora).

## 5 L'interfaccia a linea di comando

Il programma d'interfaccia (oggi chiamato `tc2`) ha evoluto gli script shell già presenti nel sistema di impaginazione FSM, automatizzando i compiti comuni che un editor compie: creare una gerarchia di directory per la rivista, creare degli stub per i singoli articoli, comporre un articolo o un intero numero, ma anche interagire con i server online per estrarre gli articoli depositati nel magazzino degli articoli disponibili (chiamato fantasiosamente Inbound di redazione), per sincronizzare la copia di lavoro con il sistema di controllo di revisione dei file o per pubblicare gli articoli o la rivista nel sistema di divulgazione attualmente utilizzato (SCRIBD).

È quest'interfaccia che costruisce il file principale L<sup>A</sup>T<sub>E</sub>X, che l'editor quindi non gestisce più direttamente, in dipendenza delle opzioni che vengono passate sulla linea di comando o in appositi file di configurazione in formato INI.

Ancora una volta l'obiettivo dello sviluppo software (poco più di 800 righe di Python) è stato quello di costruire attorno al L<sup>A</sup>T<sub>E</sub>X uno 'schermo' per permettere ad un editor, potenzialmente

incompetente su questa tecnologia, di portare a termine l'intera operazione di composizione.

La scelta si è rivelata vincente. Agli editor attuali, che pure hanno qualche conoscenza di L<sup>A</sup>T<sub>E</sub>X, molto raramente viene chiesto di accedere a quel livello per risolvere i problemi di composizione. Nell'ottica di distribuire ancor di più il ruolo dell'editor questo era un passo determinante.

A differenza dei primi tempi, oggi raramente il processo d'impaginazione produce errori nell'esecuzione dell'impaginazione L<sup>A</sup>T<sub>E</sub>X, ma quando ciò accade non sempre è facile per un editor che non lo padroneggi determinare le cause dell'errore.

## 6 I server di condivisione

Un'altra introduzione significativa nel processo di publishing, proprio per tener fede all'idea di openness che si voleva infondere al progetto è stata la condivisione del 'codice' delle riviste (non degli strumenti di programmazione, ma proprio degli articoli) attraverso un server SVN. In prospettiva anche il deposito degli articoli potrebbe essere realizzato in questa modalità.

Gli editor hanno quindi accesso alle riviste, sottodirectory di una repository SVN, e possono lavorare in autonomia sulle sezioni di una stessa rivista per poi 'committare' il loro risultato agli altri editor.

Questa non è una grande novità per i programmatori e, tra tutte, questa è stata l'innovazione sicuramente più 'amata' dagli editor, che sono essi stessi programmatori.

Essendo i file di produzione (sia gli articoli che le configurazioni) semplici file di testo, l'utilizzo di un sistema di controllo delle revisioni è perfettamente adeguato e svolge il proprio compito efficacemente ed efficientemente.

Simpaticamente anche nella realizzazione dei contenuti delle riviste si ritrovano gli stessi problemi che solitamente si hanno con il codice sorgente dei linguaggi di programmazione (es. la rottura delle 'build' o i conflitti di redazione, con i soliti conseguenti strascichi polemici).

## 7 Le macro L<sup>A</sup>T<sub>E</sub>X

I primi numeri delle riviste sono stati composti con macro L<sup>A</sup>T<sub>E</sub>X solo leggermente riviste rispetto a quelle di FSM. Si è preferito rimandare il lavoro di fine-tuning e di 'pulizia' ad un secondo momento.

In un primo tempo questo lavoro è stato condotto adattando il codice presente alla nuova struttura, senza neppure molto comprenderne il contenuto. Non sono un T<sub>E</sub>Xnico così bravo da poter comprendere un codice così complesso senza uno studio approfondito, e con le mille cose da fare per rimettere in sesto l'editore questo momento si allontanava sempre di più.

Poi quest'estate, in una di quelle notti da hacker in cui qualsiasi cosa sembra essere possibile, ho trovato la forza d'animo di compattare il codice dei pacchetti `journal` e `paper` (di cui parla più approfonditamente Gianluca in PIGNALBERI (2005a) e PIGNALBERI (2005b)), cancellando più del 30% di codice superfluo e riordinando una infinità di piccole cose impiasticciate alla men peggio precedentemente.

È nata così la classe `imediaj.cls` e il package `imediasty` per la generazione delle copertine.

Rispetto alla versione di FSM poche differenze minori, e alcune veramente sostanziali.

Tra quelle minori essenzialmente nuovi ambienti per listati, figure, zoom, pubblicità, riquadri di varia natura sia floating che in posizione di rigore, implementati in L<sup>A</sup>T<sub>E</sub>X ma richiamati da appositi markup nel file `.tchl.txt`, nonché un insieme di nuovi comandi come `tchlarticle`, `thcleditorial`, `tchltoc`, ecc., che agiscono da interfaccia per il programma principale L<sup>A</sup>T<sub>E</sub>X, e di cui parleremo dopo. È stata anche resa più flessibile l'introduzione delle testate grafiche per gli articoli e migliorata la leggibilità dell'indice, nonché gestita la numerazione a due componenti (volume, fascicolo) in luogo di quella basata sul solo numero d'uscita.

Quelle sostanziali riguardano invece:

- la 'configurazione' delle riviste;
- la gestione dei colori;
- la gestione dei font;
- la realizzazione della copertina.

## 7.1 Creazione e configurazione di una rivista

Ogni uscita della rivista è contenuta in una directory apposita con una propria gerarchia e denominata con un'etichetta relativa alla testata (CP, DEV... seguita da due cifre relative al volume e due relative al fascicolo nel volume (ad es. CP1703).

Nella directory principale del fascicolo è presente un file principale di configurazione (`CP1703.conf`), tre directory (`a/` contiene gli articoli, `c/` contiene le sezioni d'inclusione T<sub>E</sub>X previste dall'impaginazione ma dipendenti dall'uscita, e `g/` le immagini e i grafici).

All'interno della directory `a/` è presente una sottodirectory per ogni articolo denominato con la label del fascicolo e una label univoca per l'articolo, ad es. `CP1703_apollo11`.

La configurazione della rivista è ottenuta con un file in formato INI contenente una 'stanza' dedicata alla rivista, e un insieme di chiavi, obbligatorie, per indicare le informazioni variabili. Ad es:

```
[CP-1703]
volume=17
number=03
issue=1703
month=Luglio-Agosto
```

```
year=2009
```

Questi valori, ed altri letti da un file di configurazione generale, saranno riportati all'interno del file L<sup>A</sup>T<sub>E</sub>X generato automaticamente dal programma d'interfaccia `tc2` come variabili L<sup>A</sup>T<sub>E</sub>X, ad esempio:

```
%%% BASE VARS
\magazine={CP}
\headstrip={ }
\headingcolor={ }
\issuevolume=17
\issuenum={3}
\issuenumber={1703}
\issuemonth={Luglio-Agosto}
\issueyear=2009
\issnnumber={ }
%%% MAGAZINE VARS
\newtoks\varmagmagazine
\varmagmagazine={Computer Programming}
\newtoks\varmagheading
\varmagheading={CP}
%%% ISSUE VARS
\newtoks\varissuenumber
\varissuenumber={03}
\newtoks\varissuemonth
\varissuemonth={Luglio-Agosto}
\newtoks\varissueyear
\varissueyear={2009}
\newtoks\varissuevolume
\varissuevolume={17}
\newtoks\varissueissue
\varissueissue={1703}
```

Per indicare la successione degli articoli sono utilizzate le chiavi `prog<nn>`, come in:

```
prog01=cover
prog02=ad,i_0901_smartcard.png
prog03=edit
prog04=toc
prog11=focus,apollo11
prog12=focus,apollo11vs
prog41=article,abcjquery
prog42=article,stereovision
prog43=article,statmachineintro,
i_0901_programmatore_mb.png
prog80=column,kmcm
prog88=column,autori,0904_redsign_mb.png
prog89=intoc,logo.png
prog98=ad,i_0901_sinibaldi.jpg
prog99=ad,i_0901_CPweb.png
```

Queste chiavi hanno una o più componenti separate dalle virgole. La prima componente indica la "classe" dell'elemento, la seconda l'etichetta relativa.

Le chiavi principali sono quelle relative agli articoli, specificamente `focus`, per gli articoli della sezione speciale che tratta dell'argomento relativo allo strillo di copertina, `column` per le rubriche e `article` per tutti gli altri.



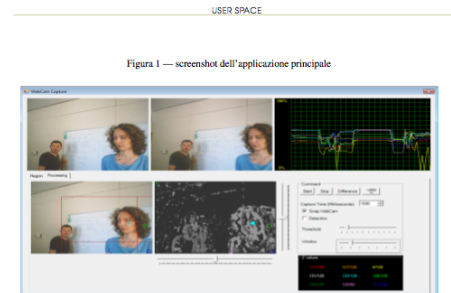


FIGURA 3: Le prime due pagine di un articolo mostrano il miglior abbinamento dei colori del titolo e del suo sfondo rispetto a quanto fatto in Free Software Magazine

Il secondo componente relativo alle chiavi utilizzate per gli articoli è l'etichetta che si troverà nella denominazione della directory. Un eventuale terzo componente indica la presenza di una mezza pagina pubblicitaria a sviluppo orizzontale o verticale a seconda che il nome dell'immagine termini con `_mb` (mezza bassa) o `_mv` (mezza verticale).

Altre chiavi sono presenti per l'impaginazione della copertina (`cover`), dell'editoriale (`edit`), del sommario (`toc`), e di pagine pubblicitarie (`ad`) o contenenti immagini a piena pagina (`img`).

Nel sorgente L<sup>A</sup>T<sub>E</sub>X le chiavi relative alla successione degli articoli saranno riportate come appositi comandi, ad esempio:

```
\tchlcover
\tchlad{i_0901_smartcard.png}{}
\tchledit{}{}
\tchltoc{}{}
\tchlfocus{apollo11}{}
\tchlfocus{apollo11vs}{}
\tchlarticle{abcjquery}{}
\tchlarticle{stereovision}{}
\tchlarticle{statmachineintro}
  {i_0901_programmatore_mb.png}
\tchlcolumn{kmcm}{}
\tchlcolumn{autori}{0904_redsign_mb.png}
\tchlintoc{logo.png}{}
\tchlad{i_0901_sinibaldi.jpg}{}
\tchlad{i_0901_CPweb.png}{}

```

Viene così completato il file L<sup>A</sup>T<sub>E</sub>X che compilato un paio di volte realizza il PDF completo della rivista.

**Stereo vision**

La Stereo Vision (visione stereo) è una tecnica per dedurre la posizione 3D degli oggetti da due o più viste contemporanee di una scena. L'uomo ha due occhi complanari posti nella parte anteriore della testa; grazie a questa particolare disposizione ravvicinata e allineata, ogni occhio ha una visione della stessa scena da un angolo leggermente diverso. I due punti di vista degli occhi sono molto simili, ma ciascuno raccoglie informazioni visive che l'altro non può raccogliere (disparità fra le scene). Ogni occhio ha la propria visuale e le due distinte immagini sono inviate al cervello per l'elaborazione: la mente umana è capace di utilizzare queste disparità per stimare la profondità della scena. La stima della disparità (in maniera rapida) è un problema complicato; le differenze possono essere causate da occlusioni di oggetti, riflessioni speculari, rumore del sensore e varie altre cause.

che riprendono la stessa scena, simulando così il sistema visivo umano. Molti modelli computazionali di visione stereo, nel matching delle corrispondenze, assumono un vincolo di unicità: ogni caratteristica individuata in un'immagine deve essere trovata in una e una sola caratteristica nelle altre immagini [1].

La Stereo Vision è una tecnica per dedurre la posizione 3D degli oggetti da due o più viste contemporanee di una scena.

Tale vincolo sembra essere plausibile, poiché negarlo, sarebbe equivalente a supporre che le caratteristiche della scena corrispondenti al matching siano in due posti contemporaneamente. Il valore del vinco-

**7.2 La gestione dei colori e dei font**

Nel sistema FSM la definizione dei colori delle differenti componenti della rivista era contenuta all'interno delle classi di documento (`journal` e `paper`) in modo rigido, ed anche un po' disordinato. Questo innanzitutto imponeva una certa quantità di acrobazie direttamente all'interno dei file L<sup>A</sup>T<sub>E</sub>X degli articoli quando, per ovvie ragioni, la composizione cromatica dei titoli mal si addiceva agli sfondi delle testate. Questo vale tanto di più nell'utilizzo fatto da Infomedia dove, a differenza di FSM, le testate sono più varie e talvolta legate esclusivamente all'articolo piuttosto che alla sezione della rivista.

Era quindi necessario passare ad uno schema in cui si potessero selezionare cromatismi differenti non solo per le differenti testate, ma da fascicolo a fascicolo della stessa testata, e addirittura da articolo ad articolo nello stesso fascicolo. Ovvero doveva essere possibile definire cromatismi differenti per qualsiasi elemento dell'impaginazione, a livello di singolo articolo (come in figura 3).

Si è risolto questo problema prima definendo un insieme completo di nomi di colori relativi alle differenti parti della rivista (sono oltre 100). E successivamente accludendo ad ogni fascicolo (nella directory `c/`) un file (ad es. `CP1703_tcccolor.tex`) che definisse prima l'insieme dei colori e poi sezioni alternative in base ad una etichetta definibile a livello di variabili d'articolo.

Seppure molto semplificato, si può vedere l'esempio successivo:

```
\newcommand{\colorscheme}[1]{
```



```

% Standard colors
\definecolor{normal}
  {rgb}{ 0, 0, 0}
\definecolor{background}
  {rgb}{ 1, 1, 1}

\IfSubStringInString\{whiteonblack}
  {#1}{%
  \definecolor{normal}
    {rgb}{ 1, 1, 1}
  \definecolor{background}
    {rgb}{ 0, 0, 0}
  }{}

\IfSubStringInString\{redonblue}
  {#1}{%
  \definecolor{normal}
    {rgb}{ 1, 0, 0}
  \definecolor{background}
    {rgb}{ 0, 0, 1}
  }{}
}

```

In questo modo richiamando nel file  $\LaTeX$  dell'articolo il comando

```
\colorscheme{whiteonblack}
```

si ottiene la modifica dei colori.

Questo mi dà modo anche di introdurre una altra peculiarità della nuova struttura. Come è stato detto in precedenza, l'editor cui spetterebbe la definizione dei colori non dovrebbe scendere al livello del  $\LaTeX$  per queste definizioni.

In realtà il sistema di traduzione `.tchl.txt`→ $\TeX$  permette la definizione di *variabili* al livello dell'articolo con il markup testuale TCHL che sono poi trasposte in variabili  $\LaTeX$ .

Ad esempio nell'intestazione del file `.tchl.txt` di un articolo è possibile rinvenire le seguenti variabili

```

#!title!Atterraggio lunare
#!author!Emmanuele Somma
#!journalpart!Cover Story
#!subtitle!Come conquistare la luna
  in soli 4 kbyte di RAM
#!colorscheme!white

```

che saranno trasformate in  $\LaTeX$  dal programma ELisp come:

```

\title{Atterraggio lunare}
\author{Emmanuele Somma}
\journalpart{Cover Story}
\subtitle{Come conquistare la luna
  in soli 4 kbyte di RAM}
\colorscheme{white}

```

In questo modo è possibile introdurre l'indicazione del colorscheme direttamente nel file TCHL.

Per ciò che riguarda la gestione dei font lo schema adottato è sostanzialmente simile.

## 8 La copertina

Per la generazione della copertina fino al penultimo fascicolo ci siamo affidati, sulle orme di FSM, all'introduzione di una pagina contenente un'immagine realizzata da un grafico. Ma in definitiva non volevamo rinunciare alla possibilità di far introdurre gli strilli dagli editor invece che dai grafici. Già dall'ultimo numero di CP — Computer Programming — è stata realizzata l'integrazione direttamente in  $\LaTeX$  della copertina completa tranne gli strilli e gli strilli generati dal  $\LaTeX$  (figura 4), in verità letti anch'essi dal file esterno di configurazione del fascicolo come:

```

[CP-1703]
...
coverlineA=ABC di jQuery
coverlineB=StereoVision
coverlineC=Traduzioni statistiche
coverlineD=FIAT in CC?
covercolor=cCOVwhite
...

```

Per rendere ancora più semplice questa composizione, e per migliorare la veste grafica è stato deciso di adottare nel prossimo futuro una linea grafica più minimalistica studiando una completa rivoluzione della copertina.

Dal punto di vista  $\LaTeX$  è stato realizzato un package apposito per la copertina (`imediacy.sty`) che, allo stato attuale non fa che comporre l'immagine di sfondo e gli strilli in posizione obbligata, ma nella nuova versione grafica permetterà l'introduzione di tutte le componenti necessarie, come overlay in un certo ordine ben definito.

A dire la verità uno dei motivi per giungere a questo risultato è anche dovuto ad un problema 'laterale' non indifferente: in questo momento non si è fatta una differenziazione della composizione in relazione alla modalità di distribuzione separando, ad esempio, la composizione per l'online da quella della stampa (che invece FSM faceva), il file PDF prodotto dalla composizione è decisamente molto pesante (30 megabyte circa), di questo il buon 30% è dovuto alla copertina realizzata dal grafico in risoluzione per la stampa. L'adozione di una grafica minimale riduce questo 'costo' ma soprattutto la ricerca dell'immagine di copertina che raramente risulta adeguata al contenuto.

## 9 Conclusioni

Il tentativo di fare un respawn di Infomedia, variando le sue caratteristiche fondamentali come struttura editoriale e modello di business, è una sfida che ci sta appassionando ogni giorno di più.



FIGURA 4: La copertina di Computer Programming Volume 17 n° 3 ha gli strilli composti con LATEX

I risultati di questi mesi sono molto evidenti, ma è ancora ben poca cosa rispetto a quello che potrebbe essere fatto da una comunità più forte e attiva.

LATEX gioca una parte determinante in questo progetto vista la forte determinazione di continuare la produzione cartacea della rivista.

In effetti lo spazio per migliorare questa componente è molto ampio viste le limitazioni e le rigidità che la contraddistinguono adesso (pur facendo un lavoro assolutamente apprezzabile ed adeguato nel frattempo).

Per avere informazioni:

- Il sito del Turing Club: <http://the.programmers.net>
- Il sito contenente il progetto TCHL, la linea di comando d'interazione tc2, il programma di traduzione Lisp e le classi e i package LATEX necessari: <https://www.assembla.com/wiki/show/turing>

- Il wiki con il materiale e le idee sulla produzione: <http://wiki.infomedia.it>
- Il blog con le comunicazioni più importanti: <http://blog.infomedia.it>
- Il nuovo sito Infomedia per la distribuzione delle riviste (accesso limitato ai sostenitori): <http://www.infomedia.it>

### Riferimenti bibliografici

- PIGNALBERI, G. (2005a). «Creating Free Software Magazine». *Free Software Magazine*, (7).
- (2005b). «Della produzione di una rivista in LATEX». In *Atti del GJTrmeeting2005*. Pisa.

▷ Emmanuele Somma  
Infomedia