

«Produrre oggetti di bellezza» nell'edizione musicologica e musicale (con un computer e la filosofia WYSIWYM)*

Piero Faustini[†]

«Invisibili fili». Musica, lessicografia, editoria e tecnologie
dell'informazione tra il XVI e XXI secolo
XI settimana di Alti Studi Rinascimentali
Ferrara, 26.v.2009

1. «Science is knowledge which we understand so well that we can teach it to a computer», affermava nel 1974 l'eccentrico matematico Donald E. Knuth, creatore di $\text{T}_{\text{E}}\text{X}$,¹ un sistema per la tipocomposizione automatica di alta qualità attraverso il quale sarebbe riuscito, pochi anni dopo, a insegnare al calcolatore i segreti tipografici in un'epoca aliena persino al concetto di personal computer. $\text{T}_{\text{E}}\text{X}$ fu la conferma indiretta della tesi di Knuth secondo la quale la programmazione stessa sarebbe un'arte, «especially because it produces objects of beauty».²

Un altro insigne matematico, Lesley Lamport, ampliò $\text{T}_{\text{E}}\text{X}$ in $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$, un sistema ad alto livello di astrazione per l'assistenza alla stesura di documenti complessi. Grazie all'impegno della comunità *open source*, $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ viene oggi utilizzato da decine di migliaia di studiosi in tutto il mondo per redigere articoli, studi e dissertazioni di matematica, scienze informatiche, fisica, ingegneria, ma anche economia e studi giuridici; e, negli

*Il testo qui riportato è in linea di massima quello esposto in occasione delle giornate di studio. Alcune puntualizzazioni aggiuntive si sono rese necessarie in seguito al dibattito e alle discussioni che si sono svolte durante il convegno e dietro le sue quinte e che hanno coinvolto tra gli altri Alessandro Roccatagliati, Pietro Zappalà e Nicola Orio. L'autore ha cercato di mantenere l'equilibrio tra esposizione teorica e più concreti esempi operativi per lo studioso interessato all'uso di sistemi WYSIWYM e simili.

[†]Università di Ferrara.

¹DONALD E. KNUTH, *Computer Programming as an Art*, «Communications of the Association for Computing Machinery» 17.12 (1974), discorso tenuto in occasione del ricevimento del Turing Award, pp. 667–673: 668.

²*Ibid.*, p. 673.

ultimi anni, dietro le teste di ponte della logica, della linguistica e della filologia, un po' tutte le discipline umanistiche.

Negli anni Ottanta si andava parallelamente affermando nel campo dei software di videoscrittura il concetto di WYSIWYG, What You See Is What You Get, 'Ciò che vedi è ciò che ottieni': finalmente, l'utente vedeva durante la stesura il risultato finale del processo stesso, come nella pagina stampata. Ciò però comportava la rinuncia a determinate istanze di qualità della composizione, al fine di visualizzare in tempo reale il risultato sullo schermo di PC migliaia di volte meno potenti degli attuali. Nonostante il successivo, vertiginoso sviluppo software e hardware, questo peccato originale della videoscrittura WYSIWYG è ancora costitutivo e caratteristico di quella mentalità "tutto e subito" con la quale le politiche dei grandi monopoli dell'informatica hanno da allora cercato di sedurre e tenere legato a sé il crescente, immenso bacino di utenti di personal computer.

In fondo, la filosofia WYSIWYG non è certo 'progressiva': dato che con Microsoft Word o alternative analoghe, magari *open source* come OpenOffice, il contenuto *corrisponde* alla pagina, il redattore è *dipendente* dal mezzo e dalla sua forma, in maniera non dissimile dai tempi di macchina da scrivere o penna e calamaio. Ciò, come vedremo, è una capitolazione delle potenzialità 'umane' dell'«informazione automatizzata»: almeno qui prevale nei fatti la *vulgata* apocalittica che vorrebbe un'umanità assillata dal costante sforzo di adeguarsi al *mezzo* (la carta o uno schermo) e non vuole invece riconoscere in molta altra tecnologia di informazione automatizzata l'evidente impegno ad assimilare, con il nostro aiuto, la mentalità umana.

2. Comunque sia, all'ora di scrivere documenti complessi ed esigenti come un volume scientifico, un articolo accademico, un'edizione critica o, come nel caso di chi vi parla, una dissertazione di dottorato di ambito musicologico, sarà fondamentale, al fine di scegliere gli strumenti giusti, cercare di capire cosa si vuole esattamente da essi.

Nell'attesa del (bel?) giorno in cui la macchina farà ricerca e scriverà la tesi di dottorato al nostro posto, essa potrebbe essere già in grado di compiere tutte le mansioni più noiose, pignole, meccaniche (in una parola: stupide) nella preparazione del testo. Le prime che ci vengono in mente sono la tipocomposizione, l'impaginazione e la produzione di un *output* adatto alla stampa o allo scambio diretto con editori o lettori. Prima ancora però verrebbero la formattazione automatica dei riferimenti interni (figure, note e rimandi) o esterni (citazioni e bibliografia), e relativa gestione. Il sistema dovrebbe permettere di personalizzare i criteri per questi ordini di esigenza, e conoscerne già i più comuni nell'ambito specifico o nella tradizione locale. Tutto ciò è il compito primario, e per eccellenza, di L^AT_EX.

Ma non sono queste le uniche esigenze che uno studioso potrebbe avanzare. Un *software* in fase di scrittura dovrebbe permettere all'utente di, parafrasando il passo citato

di Knuth, «insegnare al computer» ciò che sta scrivendo, o meglio, mostrargli di cosa si tratta, in un processo del tutto distinto dalla composizione finale.

Occorre qui una doverosa parentesi. Nella filosofia dei *database* è regola fondamentale la distinzione dei tre elementi costitutivi del sistema, ossia *structure*, *data* e *view*: ‘struttura’, ‘dati’ e ‘rappresentazione’ (oppure ‘struttura’, ‘contenuto’ e ‘forma’). Un database semplicissimo come una rubrica telefonica ha una *struttura* fissa, sempre identica: nome, cognome, numero di telefono. Il *contenuto* corrisponderà agli effettivi nominativi della *mia* rubrica. La *forma* sarà invece il come io visualizzo o modifico questi dati: in inglese l’interfaccia per modificare le informazioni in un database si chiama *form* (‘modulo da compilare’ o, appunto, ‘forma’). Il modello tripartito può, con lievi aggiustamenti, essere applicato a quegli oggetti che chiamiamo genericamente ‘documenti’, e nella fattispecie articoli, saggi o libri di carattere accademico.

Tutti i libri presentano un titolo, uno o più autori, un editore (si spera) in un frontespizio; un certo numero di capitoli suddivisi in paragrafi, capoversi, ecc.; note, riferimenti bibliografici; figure, tabelle; sommario, indice, bibliografia. Sono questi gli elementi costitutivi di una struttura, che, per non generare confusione d’ora innanzi intenderemo implicitamente come *struttura ideale* di un libro, di qualsiasi libro. Ciascun libro poi sarà ovviamente caratterizzato da differenti contenuti, o per meglio dire *contenuti strutturati* sulla base della nostra struttura ideale. E la *forma*? Un libro può avere una bellissima o bruttissima forma finale: sarà meglio, per un elaborato accademico, che abbia una forma ‘tradizionale’, e se pure sarà funzionale a una lettura comoda e alle particolari esigenze del fruitore, le probabilità che finisca per avere una forma ‘brutta’ tenderanno allo zero. Ma c’è un’*altra forma* che un libro dovrà assumere: quella con la quale, *in fieri*, sarà rappresentato al suo autore nell’atto di scriverlo con l’ausilio della macchina.

Questa forma *in fieri* (o *form!*) sarà sì una rappresentazione del contenuto, ma conveniente, qui, alla sua creazione, *non* alla sua fruizione: non v’è ragione per cui quest’interfaccia tra autore e macchina debba corrispondere alla pagina stampata. La filosofia WYSIWYG, dunque, spregiativamente ribattezzata WYSIAYG (What You See Is All You Get: ciò che vedi è tutto ciò che ottieni), dovrà lasciare il posto a un nuovo acronimo, WYSIWYM (What You See Is What You Mean: ciò che vedi è ciò che vuoi dire).

3. \LaTeX promette (e in parte mantiene) la separazione tra struttura, contenuto e forma, e la conveniente rappresentazione durante la fase di creazione, attraverso una codifica del testo spartana di tipo completamente testuale (*codice sorgente*), che ricorda quella dell’HTML o di un linguaggio di programmazione. Il *file sorgente* che la contiene, redatto anche attraverso un comune editor testuale ASCII, viene processato (*compilato*) da \LaTeX (eventualmente con il sistema \BibTeX che gestisce i riferimenti bibliografici da un *file* esterno)³ che produce un file PDF pronto per la stampa. Non c’è bisogno di alcuna GUI (interfaccia grafica): della compilazione non vedremo altro che il risultato finale.

³Il *file* bibliografico (la nostra biblioteca ‘virtuale’) andrà gestito da un software di *reference management*.

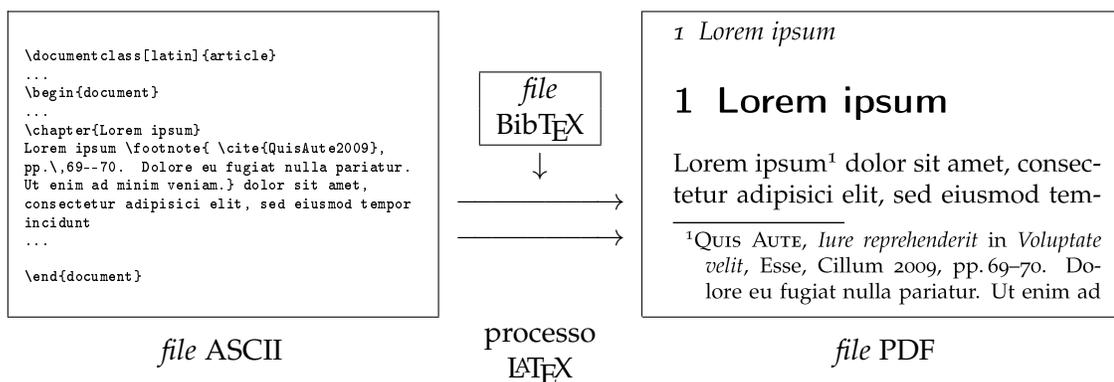


FIGURA 1: Funzionamento di \LaTeX

Nel codice sorgente, oltre al testo vero e proprio, si inseriscono anche comandi (semplici parole inglesi introdotte dal carattere «\» e seguite da un argomento tra graffe), che istruiscono \LaTeX , ad esempio, sulla natura del testo nell’argomento. Nell’esempio (fig. 1), il primo comando dichiara il documento un «articolo» in «latino», il secondo ne marca il principio vero e proprio. A un dato momento, viene dichiarato il principio del capitolo «Lorem ipsum», e a seguire il testo, nel quale è subito contenuta una nota, marcata dal comando `\footnote` e recante come argomento il relativo testo; a sua volta, la nota inizia con un riferimento bibliografico (`\cite`) che rimanda, attraverso un’etichetta univoca convenzionale, a un record del file `BibTeX` esterno contenente i dettagli bibliografici.

Nonostante il codice sorgente mantenga un minimo di leggibilità, sarebbe inutile voler negare l’aspetto quasi esoterico di \LaTeX , definito ironicamente dalla sua *community* un «segreto ben custodito».

Tuttavia, esiste un approccio più “umano” a \LaTeX , chiamato `LyX`, il primo e finora unico concreto esito genuinamente WYSIWYM nel campo dell’elaborazione di testi (o meglio, documenti).⁴

`LyX` (fig. 4 a pagina 9) ricorda un comune wordprocessor, eccetto per la rappresentazione grafica semplice e spartana: coerentemente con una rappresentazione astratta del contenuto, non esistono ‘pagine’; una nota non si trova “a piè pagina” ed è invece una sorta di scatola posizionata al posto del rimando alla nota; i riferimenti bibliografici sono selezionabili da un menù a tendina. Come in \LaTeX , sommario, indici o bibliografia semplicemente non devono essere redatti: compaiono “automagicamente” nel documento finale, magari con criteri personalizzati. Come `TeX` e \LaTeX , `LyX` è *libre e open source* (si può “aprire” per vedere come funziona, aggiustarlo o modificarlo, al contrario di molti software commerciali — o comuni elettrodomestici) ed è dotato di un’eccellente documentazione.

⁴Vedasi MARCO GUASTAVIGNA, *Ciò che vedi è ciò che vuoi significare*, «Insegnare» 7-8 (2006), URL: <http://www.noiosito.it/wysiwym.htm>.

I vantaggi di LyX rispetto ai wordprocessor WYSIWYG sono essenzialmente la qualità dell'*output*, l'agilità nello stendere e controllare i contenuti, la gestione avanzata della bibliografia,⁵ la disponibilità di centinaia di pacchetti (moduli) L^AT_EX per esigenze specifiche (linguistica, grafie particolari, ecc.), oltre che, banalmente, il costo (è *gratis*). Di inestimabile valore è invece l'accogliente ambiente internazionale della sua *community* di studiosi, prodighi di opinioni e consigli anche non strettamente tecnici.

Tuttavia LyX presenta alcuni svantaggi. Oltre a una curva di apprendimento seconda, per ripidità, al solo L^AT_EX puro, e alla necessaria mutazione paradigmatica nell'approccio alla scrittura, scoraggerà anche la limitata interazione con l'ambiente dei colleghi studiosi, degli editori e dei curatori di tradizione umanistica, che nella circolazione e condivisione di documenti è, a differenza della comunità tecnico/scientifica, molto attaccata allo standard *de facto* ossia Microsoft Word.

Ecco perché L^AT_EX e LyX sono consigliati per compiti complessi:⁶ ad esempio, le necessità del moderno filologo per una edizione critica sono soddisfatte da un pacchetto dedicato (Ledmac) che permette tra l'altro di inserire distinte fasce di apparato con formattazioni automatiche dei riferimenti (fig. 2 nella pagina seguente) ; o ancora, uno studioso italiano, Massimiliano Dominici, ha sviluppato un pacchetto (Dramatist) per la composizione di testi drammaturgici;⁷ altri ancora sono rivolti al campo che mi riguarda più da vicino: quello musicale.

4. Nonostante non si possano paragonare appieno necessità e particolarità della notazione musicale a quelle della tipocomposizione testuale, si può affermare che il software GNU LilyPond è un po' l'equivalente di L^AT_EX per la musica.

Anche nella notazione musicale si utilizzano software commerciali molto raffinati, come Finale e Sibelius, caratterizzati dall'approccio WYSIWYG. Nonostante la loro natura indubbiamente professionale (Finale *in primis*) presentano ordini di problema molto simili a quelli dei comuni wordprocessor: assenza di una netta distinzione contenuto-forma, difficoltà nell'ottenere risultati esteticamente e funzionalmente ottimali e nella gestione di documenti (musicali) complessi, formato chiuso e proprietario, elevati requisiti hardware, impossibilità di "aprire il giocattolo", cui va aggiunto il particolare del prezzo, spesso insostenibile per molte tasche.⁸

⁵LyX supporta BibL^AT_EX, sorta di BibT_EX al quadrato che permette di gestire in maniera personalizzata le regole di citazione e persino le comuni *latinitates* utilizzate dagli umanisti: *Id., op. cit., ibid., ecc.*

⁶In ANTONIO GULINO, *Lo sviluppo del software in ambito Open Source: il caso LyX*, tesi di laurea, Università degli Studi di Trento, 2000-2001, URL: <http://tomi.gulino.org/lyx/OpenSourceLyX.pdf>, p. 83, troviamo una statistica che illustra comunque una importante incidenza degli umanisti tra gli utilizzatori di LyX.

⁷Con il suo aiuto, chi scrive ha cercato di sviluppare il primo rudimentale codice per dei testi librettistici, che presentano problemi e complessità ancora maggiori.

⁸Nell'editoria professionale, inoltre, si utilizza anche il complesso e proibitivo software commerciale SCORE.

Chronicon Geldriae

Guillelmus de Berchen

De ecclesia S. Stephani Novimagensi

Nobilis itaque comes Otto imperio et dominio Novimagensi sibi, ut praefer-
tur, impignoratis et commissis proinde praeesse cupiens, anno LIII superius 1254
descripto, mense Iunio, una cum iudice, scabinis ceterisque civibus civitatis
Novimagensis, pro ipsius et inhabitantium in ea necessitate, commodo et utili-
tate, ut ecclesia eius parochialis extra civitatem sita destrueretur et infra muros
5 transferretur ac de novo construeretur, a reverendo patre domino Conrado de
Hofsteden, archiepiscopo Coloniensi, licentiam, et a venerabilibus dominis de-
cano et capitulo sanctorum Apostolorum Coloniensi, ipsius ecclesiae ab antiquo
veris et pacificis patronis, consensum, citra tamen praeiudicium, damnum aut
10 gravamen iurium et bonorum eorundem, impetravit.

Et exinde liberum locum eiusdem civitatis qui dicitur Hundisburg, de prae-
libati Wilhelmi Romanorum regis, ipsius fundi domini, consensu, ad aedifican-
dum et consecrandum ecclesiam et coemeterium, eisdem decano et capitulo de
expresso eiusdem civitatis assensu libera contradiderunt voluntate, obligantes
15 se ipsi comes et civitas dictis decano et capitulo, quod in recompensationem
illius areae infra castrum et portam, quae fuit dos ecclesiae, in qua plebanus
habitare solebat—quae tunc per novum fossatum civitatis est destructa—aliam
aream competentem et ecclesiae novae, ut praefertur, aedificandae satis conti-
guam, ipsi plebano darent et assignarent. Et desuper apud dictam ecclesiam
20 sanctorum Apostolorum est littera sigillis ipsorum Ottonis comitis et civitatis
Novimagensis sigillata.

3 p. 227 R 4 p. 97 N 6 p. 129 D 12 f. 72v M 13 p. 228 R 20 p. 130 D

2 proinde] primum D 5 ecclesia eius] ecclesia D: eius eius H extra civitatem *om.* H
infra] intra D 6 transferretur] transferreretur NH 7 Hofsteden] Hoffstede D: Hoffsteden
H Coloniensi] Colononiensi H dominis] viris H 8 Coloniensi] Coloniae H 10 iurium]
virium D 11 liberum] librum H qui] quae D Hundisburg] Hundisburch D: Hundisbrug
HMN: Hunsdisbrug R 12 regis] imperatoris D 13 et consecrandum *om.* H eisdem]
eiusdem D 15 comes] comites D dictis *om.* H 17 tunc] nunc H 18 ut...aedificandae
om. H 18–19 contiguam] contiguum M 19 apud *om.* H 20 est] et H littera] litteram
H 21 Novimagensis] Novimagii D sigillata] sigillis communita H

6–7 William is confusing two charters that are five years apart. Permission from St. Apostles' Church in Cologne had been obtained as early as 1249. Cf. Sloet, *Oorkondenboek* nr. 707 (14 November 1249): "... nos devotionis tue precibus annuentes, ut ipsam ecclesiam faciens demoliri transferas in locum alium competentem, tibi auctoritate presentium indulgemus..." 11–19 Cf. Sloet, *Oorkondenboek* nr. 762 (June 1254)

FIGURA 2: Esempio di edizione critica con L^AT_EX e il pacchetto Ledmac

Sull'esempio della grande tradizione dell'incisoria musicale otto–novecentesca, LilyPond mira alla qualità estetica e funzionale del risultato.⁹ Il software funziona compilando un file sorgente analogo al L^AT_EX, composto da caratteri ASCII che descrivono la musica da incidere con sintassi logico–simbolica:¹⁰ un esempio minimale è in fig. 3a, uno più complesso in fig. 3b nella pagina successiva, sviluppato in un ambiente grafico semi–amichevole per codice LilyPond.

È un approccio, quindi, opposto a quello di Finale: l'intervento "umano" si riduce a pure "direttive" o al limite a ritocchi formali che però, grazie ai potenti algoritmi di LilyPond, saranno minori per numero ed entità.¹¹

Come in un linguaggio di programmazione, parti di codice musicale possono costituire una variabile, ad esempio «\Soggett o», da invocare ogniqualvolta le ritroviamo in partitura, magari con opportune variazioni. La possibilità di dare, quindi, una descrizione *logica* della musica da stampare¹² può essere adattata alle particolari necessità del filologo e del trascrittore che curano un'edizione critica. Interventi editoriali di differente natura o tasso di arbitrarietà potranno essere marcati con etichette convenzionali, le quali richiamerebbero l'intervento di codice specifico (*una tantum*) per istruire il computer su quale soluzione grafica adottare per ciascuna categoria di interventi.¹³ Ragioni simili si possono addurre all'utilizzo di LilyPond per integrare scritti musicologici con elevato numero di esempi musicali dalle particolari esigenze grafiche: di enorme comodità è quindi la (parziale) integrazione di LilyPond all'interno di LyX (fig. 4 a pagina 9).

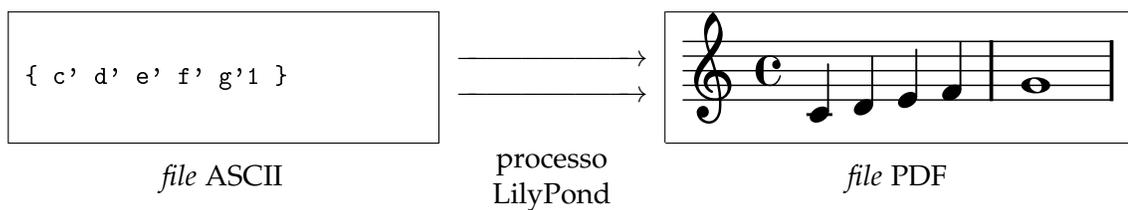
⁹LilyPond... *music notation for everyone*, URL: www.lilypond.org (visitato il 15/02/2010).

¹⁰Vedasi ERIK SANDBERG, *Separating input language and formatter in GNU Lilypond*, Tesi di Master, Uppsala: Uppsala University, 2006. Del resto, «il *software* è una rappresentazione di conoscenze specifiche di un dominio di azione, rese operative in forma logica»; AGOSTINO DI SCIPIO, *Tecnologia dell'esperienza musicale nel Novecento*, «Rivista italiana di musicologia» 35.1-2 (2000), pp. 211–246, URL: <http://xoomer.virgilio.it/adiscipi/TECN900.pdf>, p. 236.

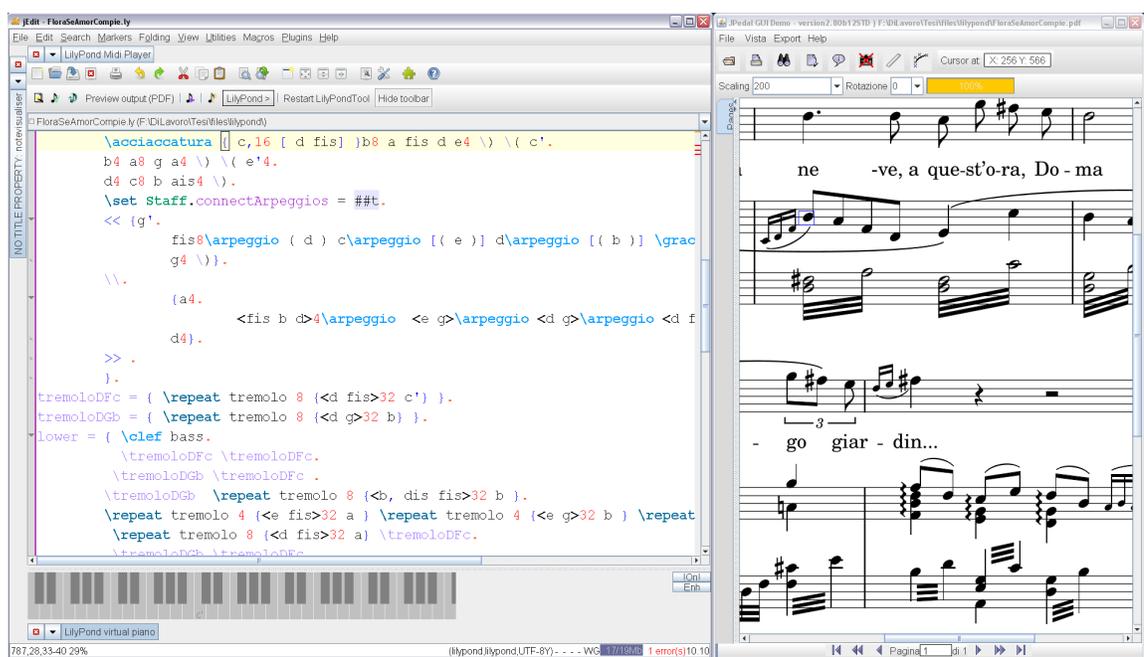
¹¹ANDREW HAWRYLUK, *Finale and LilyPond*, URL: <http://www.musicbyandrew.ca/finale-lilypond-1.html> (visitato il 15/02/2010).

¹²A titolo dimostrativo, vedasi <http://nicolas.sceaux.free.fr/prelude/prelude.html>, che illustra come produrre l'intero primo Preludio del *Clavicembalo ben temperato* di J.S. Bach a partire dalle sole note costitutive degli accordi.

¹³Rispondo a un quesito del prof. Roccatagliati sull'utilizzabilità di tale sistema per la curatela collettiva di una edizione critica di opere italiane ottocentesche, argomento tanto del suo intervento quanto di quello di Maria Chiara Bertieri. La risposta è, certamente, sì. Il vantaggio di una "edizione LilyPond" è che all'interno del suo codice possiamo lasciare traccia (sotto forma di semplice *nota*) di qualunque intervento. Inoltre, «i sistemi [di notazione] ASCII si sono caratterizzati così nel tempo per un uso sempre più orientato a fini musicologici, grazie alla loro capacità di memorizzazione e richiamo dei dati musicali che rappresentano»; MARCO RUSSO, *La notazione musicale nell'era dell'informatica*, «Nuova rivista musicale italiana» 2 (2005), pp. 209–247: 235. Di più. Attraverso le web di "programmazione condivisa" (come GoogleCode) è possibile lavorare contemporaneamente su un medesimo grande progetto: analogamente a del *codice programma*, il codice di una partitura LilyPond può essere così condiviso, e qualunque intervento registrato, catalogato e archiviato, massimizzando il coordinamento dell'*equipe*. Segnaliamo poi (in relazione all'intervento del prof. Zappalà) che LilyPond viene già utilizzato su diverse piattaforme *web* per archiviare, gestire e visualizzare frammenti e *incipit* musicali.



(A) Funzionamento di LilyPond



(B) Ambiente di lavoro LilyPondTool con codice LilyPond e output finale

FIGURA 3

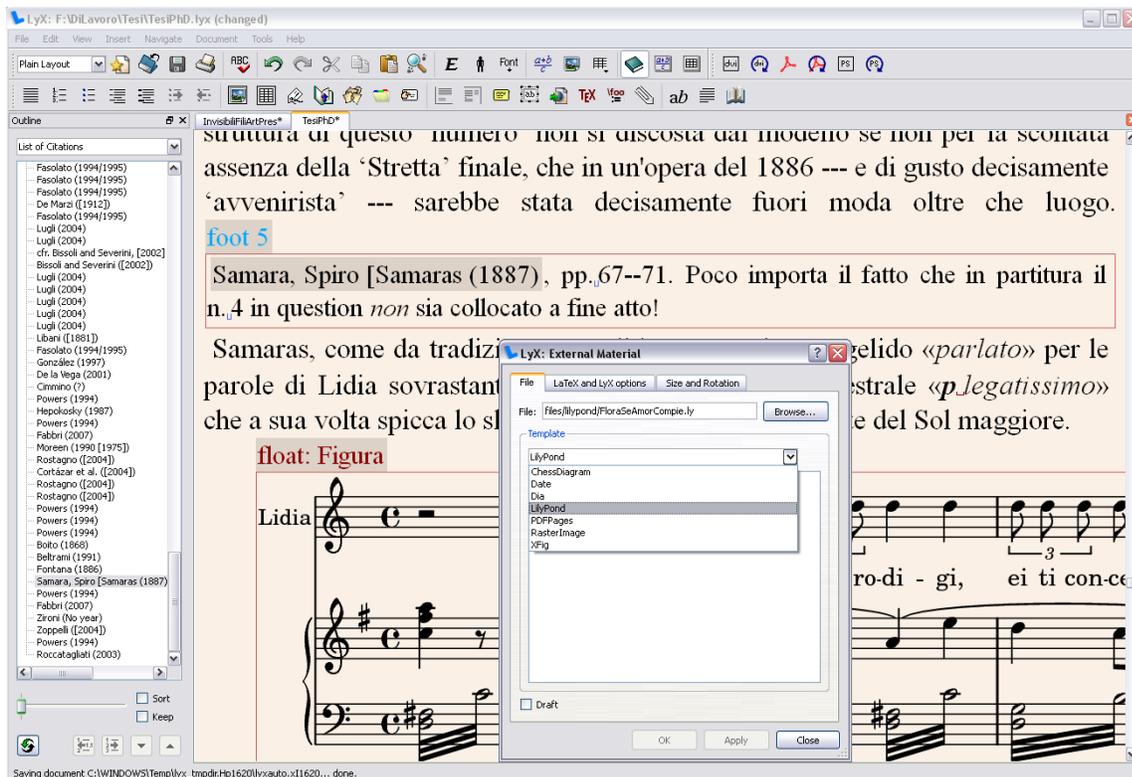


FIGURA 4: Integrazione di LilyPond in LyX

Questa breve introduzione al mondo del software per l'edizione di qualità ha avuto come principale limitazione l'ambito accademico, scientifico e filologico, ambito i cui linguaggi specifici possono essere (e, con \LaTeX e LilyPond, sono stati) insegnati al computer. Ove ciò non fosse possibile, ciascuno dovrà forgiarsi i propri strumenti perché, continuando il passo di Knuth, non si tratterà a rigore di scienza, ma piuttosto «avremo a che fare con un'arte».

Riferimenti bibliografici

- DI SCIPIO, AGOSTINO, *Tecnologia dell'esperienza musicale nel Novecento*, «Rivista italiana di musicologia» 35.1-2 (2000), pp. 211–246, URL: <http://xoomer.virgilio.it/adiscipi/TECN900.pdf>.
- GUASTAVIGNA, MARCO, *Ciò che vedi è ciò che vuoi significare*, «Insegnare» 7-8 (2006), URL: <http://www.noiosito.it/wysiwym.htm>.
- GULINO, ANTONIO, *Lo sviluppo del software in ambito Open Source: il caso LyX*, tesi di laurea, Università degli Studi di Trento, 2000-2001, URL: <http://tomi.gulino.org/lyx/OpenSourceLyX.pdf>.
- HAWRYLUK, ANDREW, *Finale and LilyPond*, URL: <http://www.musicbyandrew.ca/finale-lilypond-1.html> (visitato il 15/02/2010).
- HEWLETT, WALTER B. E ELEANOR SELFRIDGE-FIELD, cur., *The virtual score. Representation, retrieval, restoration*, Computing in musicology 12, Cambridge, Mass. Stanford, Calif.: MIT Press ; CCARH Stanford University, 2001, ISBN: 0262582090.
- HUDSON, JOHN R., *LyX, LaTeX and TeX: producing superior documents*, URL: http://www.wylug.org.uk/wp-content/uploads/2009/09/LyX-LaTeX_and_TeX.pdf (visitato il 15/02/2010).
- KNUTH, DONALD E., *Computer Programming as an Art*, «Communications of the Association for Computing Machinery» 17.12 (1974), discorso tenuto in occasione del ricevimento del Turing Award, pp. 667–673.
- *The TeXbook*, Reading, MA: Addison-Wesley, 1984.
- LilyPond... music notation for everyone*, URL: www.lilypond.org (visitato il 15/02/2010).
- LyX - The Document Processor*, URL: <http://www.lyx.org/> (visitato il 15/02/2010).
- PANTIERI, LORENZO, *L'arte di scrivere con LaTeX. Un'introduzione a LaTeX2e*, URL: http://www.lorenzopantieri.net/LaTeX_files/ArteLaTeX.pdf (visitato il 15/02/2010).
- RUSO, MARCO, *La notazione musicale nell'era dell'informatica*, «Nuova rivista musicale italiana» 2 (2005), pp. 209–247.
- SANDBERG, ERIK, *Separating input language and formatter in GNU Lilypond*, Tesi di Master, Uppsala: Uppsala University, 2006.