

Stampa musicale con **PMX**

Con una guida all'uso di **M-Tx** e all'installazione del sistema

Luigi Cataldi

WERNER ICKING MUSIC ARCHIVE

<http://icking-music-archive.org>

Rev. 3 (20 febbraio 2010)

'Hope this hilft'
Alla memoria di Werner Icking
e di Christof Biebricher

Premessa

Dalla prima edizione di questo manuale¹ sono passati quasi otto anni. I programmi si sono costantemente evoluti il **Werner Icking Music Archive** si è arricchito di tantissime nuove partiture e di molti nuovi copisti (intendo il termine con il rispetto che si deve ai filologi). Ne sono ammirato e sono lieto di aver contribuito anch'io, seppure in piccola parte, a questi progressi, magari solo per il fatto che quest'opera è stata usata come base per la documentazione introduttiva ai programmi in diverse lingue.

Una revisione era però divenuta assolutamente necessaria. Quello che presento qui ne è il risultato. Ho aggiornato le istruzioni su **PMX** e **M-Tx**; ho ricontrollato i codici degli esempi e ne ho inseriti alcuni nuovi, necessari alla descrizione delle nuove funzioni; ho aggiunto un capitolo di approfondimenti ed integrazioni al sistema e ho creato l'indice analitico. La parte più carente del testo era ad ogni modo quella relativa all'installazione, sia perché con l'aggiornamento dei sistemi anche le procedure di installazione possono cambiare, sia, soprattutto, perché rivelava la mia limitata esperienza sull'argomento. Ho anche pensato che la si poteva eliminare senza danno, ma poi mi sono ricordato delle enormi difficoltà incontrate prima di poter vedere stampata la prima partitura. Venivo da esperienze fatte di finestre e pulsantini: l'installazione fu un vero tormento. Mi sono allora convinto che, sebbene con tutti i difetti derivanti dal mio atteggiamento a volte troppo empirico, dovevo conservare questa sezione. L'ho rifatta interamente ed ora contiene le istruzioni per l'installazione su sistemi operativi Windows e Linux.

Spesso durante la correzione mi sono imbattuto in errori e ingenuità presenti nella vecchia edizione. Preoccupato controllavo la versione francese di Oliver Vogel e quasi sempre mi accorgevo che gli errori erano stati silenziosamente corretti: aggiungo riconoscenza a riconoscenza!

Ho imparato gran parte delle tecniche descritte nel capitolo 4 dalla lista di discussione di **Mu_siX_TE_X**. Molti potranno riconoscere il proprio insegnamento: ammetto i miei debiti. Quelli che ho contratto poi con gli autori di tutti i programmi qui descritti sono tanto evidenti fin dalla prima edizione che non ho bisogno di indicarli, ma solo di ribadire la mia gratitudine.

¹Uso qui il termine in modo un poco improprio, dal momento che non si tratta di un manuale di riferimento, ma piuttosto di ciò che gli inglesi chiamano 'tutorial', cioè di una guida introduttiva. Rimando ai manuali di riferimento indicati in bibliografia per una descrizione completa dei programmi.

Nel frattempo ha visto la luce ed ha avuto costanti aggiornamenti *Typesetting music with PMX* di Cornelius C. Noack, opera che, iniziata come traduzione inglese della presente, ha poi preso una strada autonoma: una vera e propria libera rielaborazione con nuovi esempi e approfondimenti della parte relativa a **PMX**. Sono lieto di avere stimolato questo lavoro e di avervi in qualche modo contribuito.

Il 17 febbraio 2009 è morto, colpito da una forma incurabile di leucemia Christof K. Biebricher, che curò la traduzione tedesca di questa guida e collaborò con numerosi contributi all'archivio **WIMA**². Fu un biofisico e un musicista. Quando conobbe la diagnosi della sua malattia disse: “Posso essere grato per avere avuto una vita interessante e bella”. Dedico alla sua memoria questa revisione.

Rinnovo i miei sentimenti di riconoscenza per tutti coloro che tengono vivo il **Werner Icking Music Archive** e ribadisco i ringraziamenti già espressi nella precedente edizione che riproduco di seguito.

Trieste, 7 febbraio 2010

Ringraziamenti (rev. 2, 2003)

Desidero ringraziare Christian Mondrup per gli incoraggiamenti e gli insegnamenti che, da quando ho iniziato a interessarmi di questi programmi, mi ha sempre fornito generosamente; tutti gli autori dei software, senza il cui lavoro nulla sarebbe stato possibile e in particolare Don Simons, non solo perché **PMX** è stata la strada attraverso la quale il mondo di **MusiX_TE_X** è divenuto a me accessibile, ma anche per la buona accoglienza e l'aiuto ricevuti da lui durante la stesura di questo manuale; tutti coloro che in vario modo partecipano all'attività del **Werner Icking Music Archive** e mia moglie Roberta per l'aiuto che mi ha fornito nella revisione del testo.

L'accoglienza alla prima pubblicazione del manuale è andata ben al di là di ogni mia favorevole aspettativa: Oliver Vogel ha curato la versione francese e Christof Biebricher quella tedesca e presto sarà disponibile anche una versione inglese. Ne sono onorato e ringrazio di cuore gli autori. Devo segnalare, poi, che nella traduzione tedesca vi sono, soprattutto nella sezione relativa a **M-T_x**, diverse aggiunte e miglioramenti rispetto all'originale italiano. Li ho accolti nella presente revisione (ne segnalo in nota la presenza) e sono doppiamente grato all'autore.

² Vedi <http://icking-music-archive.org/ByComposer/Biebricher.php>.

A proposito di questo manuale

I edizione: 22 settembre 2002

Revisioni:

1. (4 novembre 2002)
 - riscrittura del paragrafo 3.2.8, con aggiunta di un nuovo esempio musicale
 - aggiunta del presente paragrafo
 - correzione di alcuni errori di stampa
2. (21 giugno 2003)
 - aggiunta del paragrafo 3.3.7 tratto dalla traduzione tedesca di Christof Biebricher
 - modifiche apportate all'esempio 3.5 ricavate dalla traduzione tedesca di Christof Biebricher
 - suddivisione in paragrafi del capitolo relativo al testo lirico in **M-Tx**
 - aggiornamento delle istruzioni di installazione e compilazione del pacchetto postscript *tipo M*
 - correzione di alcuni errori di stampa
3. (20 febbraio 2010)
 - aggiunta dei capitoli 4 e 5 (che ora contengono anche parti di commento, codici e musiche che precedentemente si trovavano nell'Appendice); aggiunta di un nuovo codice (5.4); aggiornamento e correzione degli altri codici, in particolare di quello del paragrafo 5.3 da cui è stata ricavata la partitura urtext ed è stata condotta anche l'istruzione delle parti
 - aggiunta del paragrafo 3.2.9
 - eliminazione dell'Appendice
 - riscrittura completa dei paragrafi 1.2, 1.3 e 2.3.18
 - aggiornamento dei paragrafi: 2.2, 2.3.1 (e della figura 2.13), 2.3.4 (e della figura 2.17), 2.3.6 (con aggiunta delle figure 2.25, 2.28 e 2.29), 2.3.7 (con aggiunta delle figure 2.31 e 2.32), 2.3.10 (con aggiunta della figura 2.36), 2.3.15, 2.3.22, 2.3.24 (con aggiunta della figura 2.44), 3.2.6, 3.4.
 - sistemazione a 20 punti della grandezza del pentagramma di tutti gli esempi (escluso quello di pag. 139) e relative correzioni
 - correzione di diversi errori di stampa

Dove è reperibile: <http://icking-music-archive.org>

Dove inviare commenti: Luigi Cataldi, e-mail: *vedi* indirizzo n. 5 di figura [1.2](#), pag. [25](#).

Copyright: © 2002 –2010 Luigi Cataldi, e-mail: *vedi* indirizzo n. 5 di figura [1.2](#), pag. [25](#).

Questo documento è libero e può essere riprodotto in tutto o in parte, con ogni mezzo, purché venga mantenuto questo avviso di copyright e purché ciò avvenga senza scopo di lucro, secondo i termini della Licenza Pubblica Generica GNU, stabilita dalla Free Software Foundation.

Indice

Premessa	III
Ringraziamenti (rev. 2, 2003)	IV
A proposito di questo manuale	V
Indice	IX
Elenco delle figure	XII
Elenco delle tabelle	XIII
1 Introduzione	1
1.1 \TeX , MusiX\TeX , PMX , M-Tx	1
1.2 Installazione	3
1.2.1 Installazione sotto Windows	5
1.2.2 Installazione sotto Linux	9
1.3 Uso dei programmi	15
1.3.1 \TeX	15
1.3.2 MusiX\TeX	15
1.3.3 PMX e M-Tx	17
1.3.4 Utilizzo con le legature postscript tipo M	17
1.3.5 Compilazione automatica	19
1.4 Gli autori dei programmi	23
1.5 Il Werner Icking Music Archive	24
2 Manuale d'uso di PMX	27
2.1 Concetti preliminari	27
2.2 Preambolo	28
2.3 Corpo	32
2.3.1 Note	33
2.3.2 Pause	39
2.3.3 Accordi	39
2.3.4 Appoggiature e acciaccature	40
2.3.5 Ornamenti	41

2.3.6	Legature	43
2.3.7	Gruppi ritmici (travature)	48
2.3.8	Segni di dinamica e di espressione	51
2.3.9	Cambio di chiave	52
2.3.10	Arpeggi	53
2.3.11	Ritornelli e aste di battuta	54
2.3.12	Volta	54
2.3.13	Cambio di metro	54
2.3.14	Cambio di tonalità	55
2.3.15	Trasposizione	55
2.3.16	Titoli e testi sopra e sotto il pentagramma	57
2.3.17	Numerazione delle pagine	58
2.3.18	Comandi di carattere generale attraverso A	59
2.3.19	Spazio addizionale orizzontale nel testo musicale	61
2.3.20	Spazio minimo fra le note	62
2.3.21	Dimensioni della pagina	62
2.3.22	Comandi di impaginazione	62
2.3.23	Misure incomplete	64
2.3.24	Basso continuo	64
2.3.25	Direzione delle aste del basso	66
2.3.26	Macro	67
2.3.27	Comandi \TeX in linea	68
2.3.28	Creazione del file midi	71
2.3.29	Testo lirico	72
2.4	Limiti	72
2.4.1	Limiti sotto il diretto controllo dell'operatore	72
2.4.2	Limiti fuori dal diretto controllo dell'operatore	74
2.5	Estrazione delle parti	75
3	Breve guida all'uso di M-Tx	79
3.1	Preambolo	80
3.1.1	Omissione dei segni di battuta	84
3.2	Corpo	84
3.2.1	Misure incomplete	84
3.2.2	Linee di battuta	84
3.2.3	Numero di voci per pentagramma	85
3.2.4	Cambio di metro	85
3.2.5	Gruppi ritmici (travature)	85
3.2.6	Legature	86
3.2.7	Accordi	86
3.2.8	Segni di dinamica e di espressione	87
3.2.9	Macro	90
3.3	Testo lirico	90
3.3.1	Testo inserito fra le linee del codice	90

3.3.2	Testo inserito in un paragrafo a parte	93
3.3.3	Sinalefe e vocalizzi	95
3.3.4	Melismi	96
3.3.5	Tipi particolari di legature e impiego di <code>musixlit</code>	98
3.3.6	Controllo individuale dei versi; altri tipi di legatura	100
3.3.7	Segni di battuta per la musica antica	102
3.3.8	Due linee vocali sullo stesso pentagramma	107
3.3.9	Testo lirico sopra e sotto il pentagramma	108
3.4	Opzioni di compilazione	111
4	Approfondimenti e integrazioni	113
4.1	Comandi <code>MusiX\TeX</code> e <code>\TeX</code>	113
4.1.1	Unione di più file	113
4.1.2	Pentagrammi di diversa grandezza	114
4.1.3	Definizioni e ridefinizioni	115
4.1.4	Giustificazione verticale della pagina	119
4.1.5	Incompatibilità fra <code>M-Tx</code> e <code>scor2prt</code>	122
4.2	Trucchi per <code>PMX</code>	123
4.2.1	Fusa (1/128)	123
4.2.2	Tre voci su un solo pentagramma	124
4.3	Uso di <code>ABCPP</code>	125
4.3.1	Macro	126
4.3.2	Compilazione condizionata	126
4.3.3	Trasposizione con <code>ABCPP</code> e <code>PMX</code>	128
5	Codici e musiche	133
5.1	Dufay, <i>Kyrie</i> (<code>PMX</code>)	133
5.2	Vivaldi, <i>Coro</i> (<code>M-Tx</code>)	136
5.3	Caccini, <i>Amor l'ali m'impenna</i> (<code>ABCPP</code>)	140
5.4	Boccherini, <i>Cello concerto n. 1</i> [estr.] (<code>PMX</code>)	159
	Bibliografia	166
	Indice analitico	169

Elenco delle figure

1.1	W. A. Mozart , <i>Sonata K545</i> , batt. 1–2	2
1.2	e-mail citate nel testo	25
2.1	F. J. Haydn , <i>Quartetto Op. 76 n. 2</i> , batt. 1–4	28
2.2	Casi particolari di indicazione del metro	30
2.3	Valori del levare in 4/4	30
2.4	Chiavi	32
2.5	Claude Debussy, <i>Pelléas et Mélisande</i> , ed. Durand, pag. 149	33
2.6	Indicazione dell’ottava con valori assoluti	34
2.7	Indicazione relativa dell’ottava	34
2.8	Caratteristiche delle note	35
2.9	Posizione dei punti di valore	35
2.10	Posizione delle alterazioni	36
2.11	Codificazione semplificata dei ritmi puntati e in rapporto di 2:1	36
2.12	Gruppi irregolari di note	37
2.13	Gruppi irregolari di note: controllo del numero da stampare.	38
2.14	Gruppi irregolari di note: casi particolari.	38
2.15	Pause	39
2.16	Accordi	40
2.17	Appoggiature e acciaccature	41
2.18	Ornamenti	42
2.19	Legature	44
2.20	Limiti delle legature in MusiX_TE_X	44
2.21	Le legature di figura 2.20 con il pacchetto PS <i>Tipo K</i>	45
2.22	Le legature di figura 2.20 con il pacchetto PS <i>Tipo M</i>	45
2.23	Le legature di figura 2.19 con il pacchetto PS <i>Tipo K</i>	45
2.24	Le legature di figura 2.19 con il pacchetto PS <i>Tipo M</i>	46
2.25	Opzioni di controllo dell’arco nelle legature <i>Tipo K</i>	46
2.26	Legature di valore, MusiX_TE_X e <i>Tipo M</i>	47
2.27	Legature di valore, <i>Tipo K</i>	47
2.28	Altre opzioni di controllo nelle legature <i>Tipo K</i>	47
2.29	Effetto di Ap1 nelle legature <i>Tipo K</i>	48
2.30	Gruppi ritmici	50
2.31	Gruppi ritmici non riconosciuti automaticamente	50

2.32	Gruppi ritmici: funzione di ripetizione	50
2.33	Gruppi ritmici su pentagrammi diversi	50
2.34	Indicazioni dinamiche	52
2.35	Cambio di chiave	53
2.36	Arpeggi	53
2.37	Ritornelli e volta	54
2.38	Cambio di tonalità	55
2.39	Scala di re minore non trasposta	56
2.40	Scala di re minore trasposta di un tono	57
2.41	Scala di re minore trasposta di un semitono	57
2.42	Titoli e altre indicazioni testuali	58
2.43	Cifratura del basso	65
2.44	Cifratura del basso: caratteri speciali	66
2.45	Macro	67
2.46	Comandi \TeX in linea (tipo 1)	69
2.47	Comandi \TeX in linea (tipo 4)	70
3.1	Segni di dinamica e di espressione in M-Tx	88
3.2	J. S. Bach , <i>Passione secondo Giovanni</i> , n. 40	91
3.3	Sinalefe e vocalizzi	96
3.4	Melismi	97
3.5	Legature senza vocalizzi e impiego di musixlit	98
3.6	Manipolazione individuale dei versi	100
3.7	Segni di battuta fra i pentagrammi	103
3.8	Uso dei segni di battuta nella musica antica	105
3.9	Più voci sullo stesso pentagramma	109
3.10	Voci sopra e sotto il pentagramma	110
4.1	Ridefinizione di comandi \TeX	116
4.2	Inserimento di fuse in PMX	123
4.3	Pentagramma con tre voci in PMX	125
4.4	J. S. Bach , <i>Toccata e fuga in re minore</i> , incipit	127
4.5	J. S. Bach , <i>Toccata e fuga in re minore</i> , incipit: file midi	128
4.6	Trasposizione con ABCPP : partitura da stampare	130
4.7	Trasposizione con ABCPP : part. per midi ed estrazione delle parti	130
4.8	Trasposizione con ABCPP : parte 1	130
4.9	Trasposizione con ABCPP : parte 2	130
4.10	Trasposizione con ABCPP : parte 3	130

Elenco delle tabelle

1.1	Installazione di MusiX\TeX (Windows)	7
1.2	Installazione di PMX (Windows)	8
1.3	Installazione di M-Tx (Windows)	9
1.4	Installazione di Musixpss (Windows)	9
1.5	Installazione di MusiX\TeX (Linux)	12
1.6	Installazione di PMX (Linux)	13
1.7	Installazione di M-Tx (Linux)	14
1.8	Installazione di Musixpss (Linux)	14
2.1	Casi particolari di indicazione del metro	30
2.2	Riepilogo dei valori del <i>preambolo</i>	31
2.3	I valori numerici del preambolo di figura 2.1	31
2.4	Ornamenti	42
2.5	Simboli midi	71
2.6	Strumenti General Midi	73
3.1	Comandi del preambolo di M-Tx	81

Capitolo 1

Introduzione

1.1 $\text{T}_{\text{E}}\text{X}$, MusiX $\text{T}_{\text{E}}\text{X}$, PMX, M-Tx

MusiX $\text{T}_{\text{E}}\text{X}$ è indubbiamente uno dei migliori programmi di tipografia musicale esistenti: produce stampe di eccellente qualità, permette una facile conversione della partitura in postscript e in PDF, è consolidato e stabile, viene continuamente aggiornato ed è gratuito. Tuttavia la sua diffusione si limita, con poche eccezioni, alla ristretta cerchia dei musicisti che vengono dal mondo della scienza, poiché l'uso non è a prima vista intuitivo e qualche volta richiede la conoscenza di nozioni che sono elementari per i programmatori di software, ma ardue per i semplici musicisti. Inoltre non è WYSIWYG (What You See Is What You Get, cioè, quello che vedi è quello che ottieni). Significa che non è possibile vedere la partitura così come verrà stampata mentre la si scrive. Il sistema infatti non è interattivo. Per stampare un documento bisogna prima scriverlo in un normale file di testo usando qualsiasi editor (ad esempio **Edit** o **Notepad**), poi compilarlo con $\text{T}_{\text{E}}\text{X}$ in modo da ottenere un altro file che, con un apposito programma, può essere letto, visualizzato sullo schermo, inviato alla stampante ed anche tradotto in formato postscript per gli usi professionali. La natura simbolica e non iconica del programma è destinata a scoraggiare molti, ma è anche, in un certo senso, un pregio, poiché fa in modo che si abbia consapevolezza dell'intero procedimento di simbolizzazione e stampa della musica, che ci si concentri sul risultato tipografico finale e non si sia distratti dal semplice gioco dell'interattività.

MusiX $\text{T}_{\text{E}}\text{X}$ è un'insieme di 'macro' che permette di stampare musica con $\text{T}_{\text{E}}\text{X}$, il prestigioso programma di tipografia elettronica creato da Donald Knuth, utilizzato soprattutto per i testi scientifici. Knuth volle che il programma fosse di libera e gratuita distribuzione, di conseguenza quasi tutte le applicazioni che impaginano testi mediante $\text{T}_{\text{E}}\text{X}$ sono e resteranno assolutamente di pubblico dominio e di libero uso.

Una distribuzione di $\text{T}_{\text{E}}\text{X}$ è una specie di sistema operativo, contenente i programmi necessari per le operazioni appena viste e quelli utili per moltissime altre applicazioni. Vi sono distribuzioni per ogni sistema operativo. **$\text{T}_{\text{E}}\text{X}$ Live** o **te $\text{T}_{\text{E}}\text{X}$**

per Linux possono essere installate insieme al sistema operativo o successivamente senza problemi. Per Windows esiste una distribuzione, **MikTeX**, di facile installazione. Per Macintosh sono disponibili **CMacTeX** (<http://www.kiffe.com/cmactex.html>) e **i-Installer** (<http://www.rna.nl/tex.html>). Esiste poi **Fink**, (<http://www.finkproject.org>) un sistema che rende disponibile pacchetti Linux pronti da scaricare con il sistema di Debian `apt-get` e da usare su Macintosh, fra cui **teTeX**.

Codificare una partitura nella lingua di **MusiXTeX** resta tuttavia una cosa difficile. Fortunatamente esistono due preprocessori **PMX** e **M-Tx** in grado di semplificare drasticamente il processo di introduzione della musica: i sistemi di codificazione della partitura di **PMX** e di **M-Tx** sono fra i più semplici ed efficienti fra quelli esistenti.

Le prime due battute della *sonata K545* di Mozart possono dare un'idea dei tre diversi sistemi di codificazione (**MusiXTeX**, **PMX** ed **M-Tx**). La musica riprodotta in figura 1.1 può essere ottenuta indifferentemente con uno dei tre seguenti sistemi.



FIGURA 1.1 – W. A. Mozart, *Sonata K545*, batt. 1–2

MusiXTeX

```
\input musixtex
\parindent10mm
\setname1{Piano}
\setstaves12
\generalmeter{\meterfrac44}
\nobarnumbers
\startextract
\Notes\ibu0f0\qb0{cge}\tbu0\qb0g|\hl j\en
\Notes\ibu0f0\qb0{cge}\tbu0\qb0g|\ql l\sk\ql n\en
\bar
\Notes\ibu0f0\qb0{dgf}|\qlp i\en
\notes\tbu0\qb0g|\ibb1j3\qb1j\tb1l\qb1k\en
\Notes\ibu0f0\qb0{cge}\tbu0\qb0g|\hl j\en
\endextract
\end
```

PMX

```

2 1 4 4 4 4 0 0
1 1 20 0.12
Piano
tt
./
w120m

% Batt. 1-2
c8 g+ e g c- g+ e g | d g f g c- g+ e g Rb /
c2+ e4 g | bd4- c1 d c2 /

```

M-Tx

```

Style: piano
Piano: Voices MD MS; Clefs G G; Continuo
Name: Piano
Meter: 4/4

```

```

%% w120m
c2+          e4      g      | b4d-  c1 d c2          |
c8+ g+ e g c- g+ e g | d g f g      c- g+ e g |

```

Sebbene non sia molto più lungo, il codice **MusiX_{TeX}** è indubbiamente meno intuitivo e più complicato degli altri due.

Nel presente manuale verrà descritto dettagliatamente l'uso di **PMX**, più sommariamente quello di **M-Tx**. Saranno inoltre forniti alcuni suggerimenti per un uso più avanzato del sistema e una guida all'installazione dei programmi. Ai meno esperti la procedura di installazione potrà apparire difficoltosa. Non è il caso di scoraggiarsi: la fatica iniziale sarà ripagata con uno dei migliori e dei più semplici sistemi di tipografia musicale esistenti.

1.2 Installazione

L'installazione del sistema, per chi vi si avvicini per la prima volta, può essere difficoltosa e addirittura frustrante: deve essere fatta a mano, può variare a seconda delle esigenze personali e dei sistemi presenti sul computer. Le procedure che propongo in queste pagine, non sono le uniche possibili e può darsi che non siano le migliori, ma intendono essere semplici, pratiche e utili soprattutto ai neofiti. Per informazioni più dettagliate è comunque sempre bene fare riferimento alle istruzioni incluse nei singoli programmi. Si suppone che il lettore sappia:

- decomprimere e copiare file e directory
- lanciare un programma da riga di comando

- scrivere un file di solo testo con un editor (ad esempio `edit` per Windows o `gedit` per Linux)

In questo paragrafo saranno descritti i seguenti argomenti:

- predisposizione del sistema per l'installazione dei programmi musicali
- installazione di **MiKTeX** per Windows.¹
- installazione di **MuSiXTeX**
- installazione di **PMX**
- installazione di **M-Tx**
- installazione di **Musixpss**²

Si assume invece che nel sistema siano già presenti **GhostScript** e **GSview** (<http://pages.cs.wisc.edu/~ghost/>), oppure i corrispondenti programmi Linux³ per la visualizzazione dei file postscript e pdf.

Nelle distribuzioni di TeX tutti i componenti sono collocati all'interno dell'albero principale denominato `texmf`, che possiede una struttura standard di sotto-directory, denominata TDS (TeX Directory Structure). Se, dopo avere installato il sistema principale, aggiungiamo **MuSiXTeX** in questo stesso albero⁴, ogni volta che si reinstalla MiKTeX, ad esempio dopo un aggiornamento, si dovrà ripetere interamente l'installazione manuale di **MuSiXTeX**. Se invece usiamo un albero locale basterà conservare una copia di backup da copiare così com'è e modificare solo i file di configurazione per ripristinare il sistema di stampa musicale. Ecco dunque come procederemo:

- Windows
 - Tutti gli eseguibili (`.exe`, `.bat` o simili) saranno posti in una directory a sé stante non appartenente all'albero principale MiKTeX.
 - Tutti gli altri componenti (font, file di configurazione, documentazione, ecc.) saranno posti in un albero locale di MiKTeX
- Linux
 - Tutti i componenti (eseguibili, font, file di configurazione ecc.) saranno posti nella home directory personale.⁵

Per testare il sistema dopo l'installazione si veda il paragrafo 1.3.

¹Sotto Linux, come s'è detto, non vi sono difficoltà per installare **TeXLive** o **teTeX**. Per Macintosh suggerisco di installare **Fink** e **teTeX** e poi seguire le istruzioni per l'installazione dei programmi musicali fornite per Linux.

²Pacchetto per legature postscript di Hiroaki Morimoto (detto tipo M dal nome dell'autore) alternativo a quello di Stanislav Kneifl (tipo K) già compreso nella distribuzione di **MuSiXTeX**.

³**KGhostView**, `gv`, `evince`, `xpdf`, `kpdf` o simili.

⁴Nella prima versione di questo manuale era la procedura da me proposta, più, lo ammetto, per inesperienza che per altre ragioni.

⁵Vantaggi: non è necessario avere privilegi di root per l'installazione; tutto il sistema può

1.2.1 Installazione sotto Windows

Preliminari

1. Creiamo una directory in cui porre tutti gli eseguibili, ad esempio `c:\mieibat`.
2. Aggiungiamo la directory appena creata al `path` per poter lanciare da qualsiasi posizione in cui ci si trovi i programmi là presenti. Per ottenere ciò bisogna andare in Start-Pannello di controllo, selezionare Proprietà del sistema, poi Avanzate infine Variabili d'ambiente. Nella scheda Variabili di sistema selezionare `path` e premere `edit`. Aggiungere la directory `c:\mieibat` alla lista di quelle esistenti assicurandosi che sia separata dalle altre mediante un punto e virgola. Salvare ed uscire.
3. Creiamo un albero locale di \TeX , `c:\localtexmf`, con la seguente struttura:

```

localtexmf\doc                (per la documentazione)
localtexmf\dvipdfm\config      (per i file di configurazione di dvipdfm)
localtexmf\dvips\config        (per i file di configurazione di dvips)
localtexmf\fonts\source\public\musixtex      (per le fonti mf)
localtexmf\fonts\tfm\public\musixtex          (per le fonti tfm)
localtexmf\fonts\type1\public\musixtex        (per le fonti postscript)
localtexmf\metapost\musixtex  (per i file di configurazione di metapost)
localtexmf\tex\generic\musixtex (per tutte le macro musicali)

```

Mik \TeX

Si tratta della più diffusa distribuzione del sistema \TeX per Windows (2000 o superiore). La procedura di installazione è automatica, quindi non richiede spiegazioni particolari. Rimando comunque alle informazioni presenti sul sito. Basterà qui dire che sono disponibili due versioni: **basic** contenente solo i componenti essenziali del sistema o **complete**. Suggesto di installare la **basic**, che richiede minori risorse e minor tempo di scaricamento rispetto all'altra. \TeX è un sistema di tipografia elettronica che può fare moltissime cose diverse, dalla matematica alla musica, dalla chimica alla filologia. Ad ognuno di questi scopi è dedicata un'estensione particolare (cioè un gruppo di macro). **Mik \TeX** consente l'installazione automatica di queste estensioni. Per non appesantire il sistema conviene aggiungere quelle non presenti nella versione **basic** solo quando se ne sente l'effettiva necessità. Nella versione **complete** è presente anche **MusiX \TeX** , ma di solito non è aggiornato e non contiene tutti i programmi necessari alla stampa musicale. È meglio dunque effettuare l'installazione manuale.

essere modificato in qualsiasi momento in base alle proprie esigenze; in caso si debba reinstallare il sistema, sarà sufficiente fare un backup dell'albero \TeX della home directory e aggiornare i file di configurazione per ripristinare il sistema di stampa musicale. Svantaggi: i programmi saranno disponibili solo per l'utente che li ha installati e il per suo gruppo.

Ecco i passi, da effettuarsi con il collegamento internet attivo, per l'installazione di **MiKTeX**:

1. scaricare il programma per l'installazione della versione **basic** dal sito di MiKTeX (<http://www.miktex.org/>)
2. accettare i termini della licenza
3. scegliere se installare per ognuno che usi il computer o solo per l'amministratore
4. scegliere il luogo in cui saranno posti i programmi (si può accettare la proposta predefinita)
5. setting:
 - scegliere formato carta predefinito
 - scegliere il modo in cui avverranno gli aggiornamenti (scelta consigliata: 'ask me first')
6. riepilogo delle impostazioni scelte
7. avvio dell'installazione: i componenti scelti saranno scaricati dal sito e installati, poi il sistema sarà automaticamente configurato e risulterà pronto per l'uso.

MusiXTeX

Il pacchetto è composto da macro (*.tex), un'estensione per L^ATeX (musixtex.sty⁶), fonti bitmap (mf e tfm) e postscript (pfb), file di configurazione (*.map, *.pro) e documentazione (*.pdf e sorgenti *.tex). Ciascuna di queste parti va copiata in un luogo dell'albero locale che abbiamo precedentemente creato, `\localtexmf`, in cui possano essere trovate e riconosciute da TeX. Gli eseguibili andranno invece in `c:\mieibin`.

Dal **Werner Icking Music Archive** (<http://icking-music-archive.org>). Scaricare il file `musixtex.zip`. La normale distribuzione di **MusiXTeX** (`musixtex.zip`) una volta decompressa in una directory temporanea (diciamo `musixtex`) ha la struttura indicata nella colonna di sinistra della figura 1.1, che mostra in quale punto dell'albero `localtexmf` tali file vadano copiati.

Ora bisogna segnalare a MiKTeX i cambiamenti apportati al sistema. Per prima cosa è necessario includere il nuovo albero locale (`c:\localtexmf`) nel path di **MiKTeX**. Per far ciò si deve andare in **Start-MiKTeX 2.8-Maintenance-Settings** (che corrisponde a **MiKTeX Options**⁷), selezionare **Roots**, premere **Add** e selezionare la directory `c:\localtexmf` da includere. Poi confermare, chiudere e uscire.

Le informazioni necessarie per far sapere a MiKTeX quali fonti postscript tipo 1 devono essere usate nella compilazione al posto di quelle bitmap si trovano nei file con estensione `.map`. Per aggiornare il sistema affinché possano essere usati i file `.map` appena installati bisogna aprire una finestra dos e digitare:

```
initexmf --mkmaps
```

⁶Normalmente per stampare musica si utilizza solo TeX. L^ATeX si impiega per includere brani più corti di una pagina all'interno di documenti di testo.

⁷D'ora in poi indicheremo con **MiKTeX Options** questa scheda.

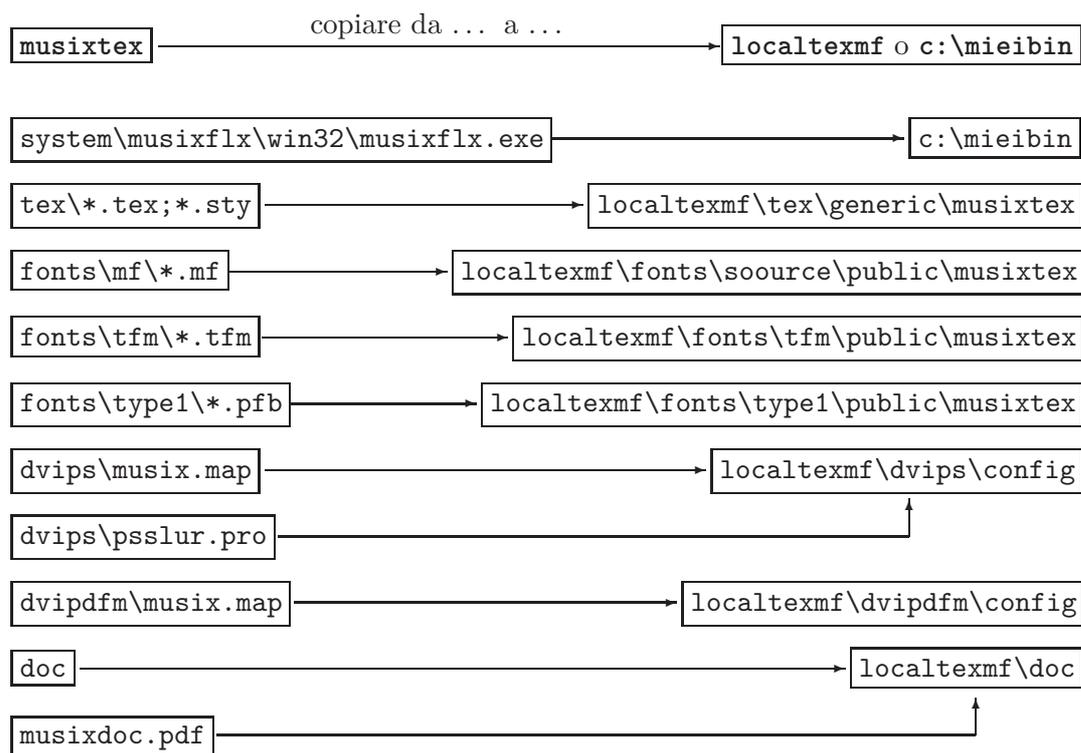


TABELLA 1.1 – Installazione di MusiXTeX (Windows)

poi

updmap

L'aggiornamento del database dei nomi dei file installati è necessario ogni volta che si aggiunge una nuova macro, affinché questa possa essere trovata e utilizzata durante la compilazione. Nel nostro caso, dobbiamo segnalare al sistema la presenza delle nuove macro musicali. Per far ciò si usa il comando

```
initexmf --update-fndb
```

o, se si preferisce l'interfaccia grafica, si deve andare in **MiKTeX Options** e avviare prima **Refresh FNDB** (update file names database) e poi **Update format**. L'installazione è completa⁸.

PMX

Il preprocessore **PMX** è essenzialmente composto da due programmi `pmxab.exe` (il preprocessore), `scor2prt.exe` (per l'estrazione delle parti) e un file `tex pmx.tex`.

⁸Il pacchetto per l'inserimento del testo lirico **Musixlyr** di Rainer Dunker, le fonti postscript e il pacchetto per ottenere segni di legatura postscript tipo K sono ora inclusi nella distribuzione principale e non necessitano di una installazione a parte.

Scaricare il file `pmx250.zip` (o una versione successiva) dalla sezione software del **Werner Icking Music Archive** (<http://icking-music-archive.org>) e scompattarlo in una directory temporanea (es.: `pmx`). Ne risulterà la struttura presentata nella colonna di sinistra della figura 1.2, nella quale sono anche indicate le collocazioni in cui i file dovranno essere copiati.

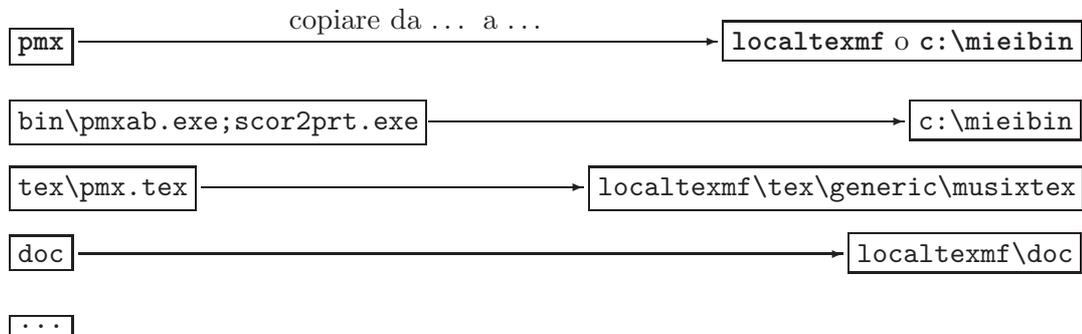


TABELLA 1.2 – Installazione di **PMX** (Windows)

In `example` ci sono esempi musicali che possono essere copiati in una directory di lavoro (es.: `c:\musica`) e utilizzati per testare il funzionamento del programma o per vederne le potenzialità d'uso.

Poi Refresh FNDB e Update format. L'installazione è ora completa e la directory temporanea può essere cancellata.

M-Tx

Gli elementi essenziali sono l'eseguibile `prepmx.exe` e `mtx.tex`.

Dal **Werner Icking Music Archive** (<http://icking-music-archive.org>) scaricare l'archivio `mtx060-win32.zip` e decomprimerlo in una directory temporanea (es.: `mtx`). Nella stessa directory scaricare anche `mtx060.pdf` (la documentazione). Copiare i file come indicato in figura 1.3.

Poi Refresh FNDB e Update format. L'installazione è ora completa e la directory temporanea può essere cancellata.

Pacchetto per le legature PostScript tipo M

È richiesto il pacchetto **MetaPost**, peraltro già presente nella distribuzione di **MikTeX**. Scaricare il file `musixpss-0.52.tar` (o una versione successiva) dal sito <http://homepage1.nifty.com/kuuku/pub/musixtex/musixpss/index-e.html>. Decomprimere l'archivio in una directory temporanea (es.: `musixpss`). Copiare come indicato nella figura 1.4.

Poi Refresh FNDB e Update format. L'installazione è ora completa e la directory temporanea può essere cancellata.

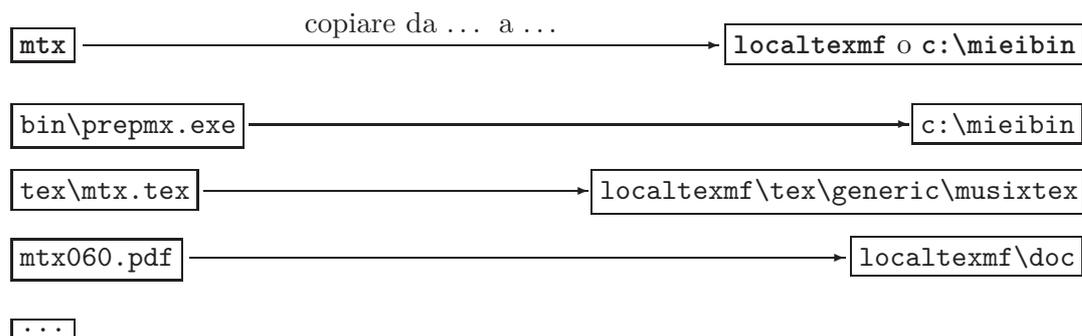


TABELLA 1.3 – Installazione di M-Tx (Windows)

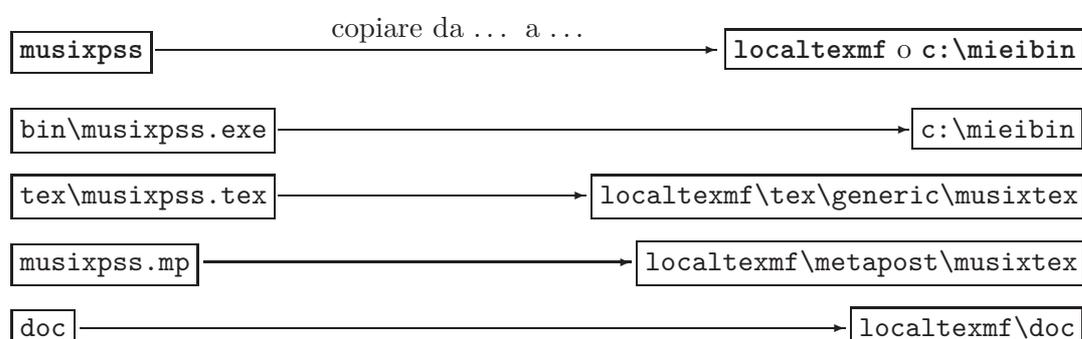


TABELLA 1.4 – Installazione di Musixpss (Windows)

1.2.2 Installazione sotto Linux

Preliminari

Installeremo, come s'è detto, il sistema nella home directory⁹ e **porremo tutti gli eseguibili nella directory /bin**, già esistente. Per essere certi che il nostro sistema \TeX sia predisposto per riconoscere l'albero collocato nella directory personale si può usare il comando

```
kpsewhich -expand-var=\$TEXMFHOME
```

oppure, se il precedente non dà esito

```
kpsewhich -expand-var=\$HOMETEXMF
```

⁹La si indica convenzionalmente come `$HOME` (sistema che impiegheremo anche qui). Ad esempio se il nome dell'utente è `mario` la directory sarà `/home/mario`. Un'altra abbreviazione comunemente usata allo stesso scopo è `~`.

In risposta sullo schermo dovrebbe apparire l'indirizzo dell'albero $\text{T}_{\text{E}}\text{X}$ personale, cioè qualcosa di simile a

```
/home/username/texmf
```

Ciò significa che l'albero personale $\text{T}_{\text{E}}\text{X}$ è già incluso nel file di configurazione, cioè che in `texmf.cnf` è presente la seguente linea

```
TEXMFHOME = \${HOME}/texmf
```

e che possiamo procedere all'installazione. In caso contrario è necessario localizzare il file di configurazione con il comando `kpsewhich texmf.cnf` e aggiungervi la linea sopra indicata, cosa per la quale si devono avere i privilegi di root. Attenzione però, nelle versioni più recenti il file `texmf.cnf` è creato automaticamente da `update-texmf` o altri simili programmi. Attenersi alle istruzioni della propria distribuzione $\text{T}_{\text{E}}\text{X}$ per effettuare la modifica.

Nel caso non si disponga dei privilegi di root è possibile aggirare il problema in questo modo. Per prima cosa bisogna fare una copia del file `texmf.cnf` da collocare nella home directory. Il posto più adatto è `!HOME/texmf/web2c`, dopodiché è però necessario includere la directory nel path della shell che si usa (consultare la documentazione specifica per questo). Una soluzione meno elegante che dovrebbe ugualmente funzionare è collocare il file di configurazione in `!HOME/bin`. Ora è necessario aprire il file con un editor. Aggiungere, se non esiste, la linea

```
HOMETEXMF = $HOME/texmf
```

oppure (i comandi sono analoghi)

```
TEXMFHOME = $HOME/texmf
```

e modificare la lista degli alberi `texmf` in modo che l'albero posto nella home directory sia indicato al primo posto come nell'esempio che segue, in cui la linea originale è conservata ma disattivata con il carattere di commento (%).

```
%TEXMF = !!$TEXMFLOCAL,!!TEXMFMAIN
TEXMF = {$TEXMFHOME,!!$TEXMFLOCAL,!!TEXMFMAIN}
```

Fatto ciò e lanciato il comando `texhash` il sistema dovrebbe funzionare prendendo le informazioni dal file di configurazione posto nella home directory anziché da quello posto nell'albero principale. Possiamo dunque proseguire con l'installazione.

Creiamo o aggiorniamo **l'albero personale** di $\text{T}_{\text{E}}\text{X}$, nella directory home in modo che abbia la seguente struttura:

```
/texmf/doc                (per la documentazione)
/texmf/dvipdfm/config     (per i file di configurazione di dvipdfm)
/texmf/dvips/config       (per i file di configurazione di dvips)
```

```

/texmf/fonts/source/public/musixtex          (per le fonti mf)
/texmf/fonts/tfm/public/musixtex            (per le fonti tfm)
/texmf/fonts/type1/public/musixtex          (per le fonti postscript)
/texmf/metapost/musixtex                    (per i file di configurazione di metapost)
/texmf/tex/generic/musixtex                 (per tutte le macro musicali)

```

Ora è possibile procedere all'installazione del sistema musicale. Nei passaggi che seguono si suppone che il sistema sia dotato dei compilatori necessari¹⁰.

MusiX \TeX

Dopo aver scaricato e decompresso l'archivio `musixtex.zip` (vedi pag. 6), collocarsi nella directory `system/musixflx/c-source`. Compilare il programma¹¹ con il comando

```
gcc musixflx.c -o musixflx
```

Poi copiare i file nell'albero personale come indicato nella figura 1.5.

È ora necessario far sapere al sistema quali fonti postscript tipo 1 vanno usate al posto di quelle bitmap. Come si è già detto le informazioni necessarie si trovano nei file con estensione `.map`. Per segnalare al sistema la presenza dei nuovi file di mappatura bisogna usare il comando

```
updmap --enable Map !HOME/texmf/dvips/config/musix.map
```

Infine si deve rigenerare il database dei nomi dei file presenti per aggiungervi le nuove macro musicali con `texhash` oppure con `mktexlsr` (i due comandi sono equivalenti).

PMX

La versione Linux di **PMX** deve essere compilata dai sorgenti Fortran. Scaricare l'ultima versione del programma (attualmente la 2.5) dal **Werner Icking Music Archive** (<http://icking-music-archive.org>). Decomprimere l'archivio `pmx250.zip` e porsi nella directory `src`. Attenzione: è necessario che i sorgenti abbiano la terminazione di linea adeguata al sistema operativo in uso, cioè nel nostro caso Unix. Per questo si può usare l'opzione `-a` in fase di decompressione dell'archivio (`unzip -a pmx250.zip`), oppure, dopo aver decompresso l'archivio nel modo consueto, si può usare l'utilità `dos2unix nomefile` per tutti i file sorgenti presenti nell'archivio. Copiare `pmxab.for` in `pmxab.f`:

```
cp pmxab.for pmxab.f
```

Fare la stessa operazione con `scor2prt.for`:

¹⁰Cioè `gcc`, `f2c`, `fpc`.

¹¹Nel caso che la compilazione non andasse a buon fine, nella distribuzione è presente una versione di `musixflx` già compilata per diversi OS, fra cui Unix, pronta per l'uso.

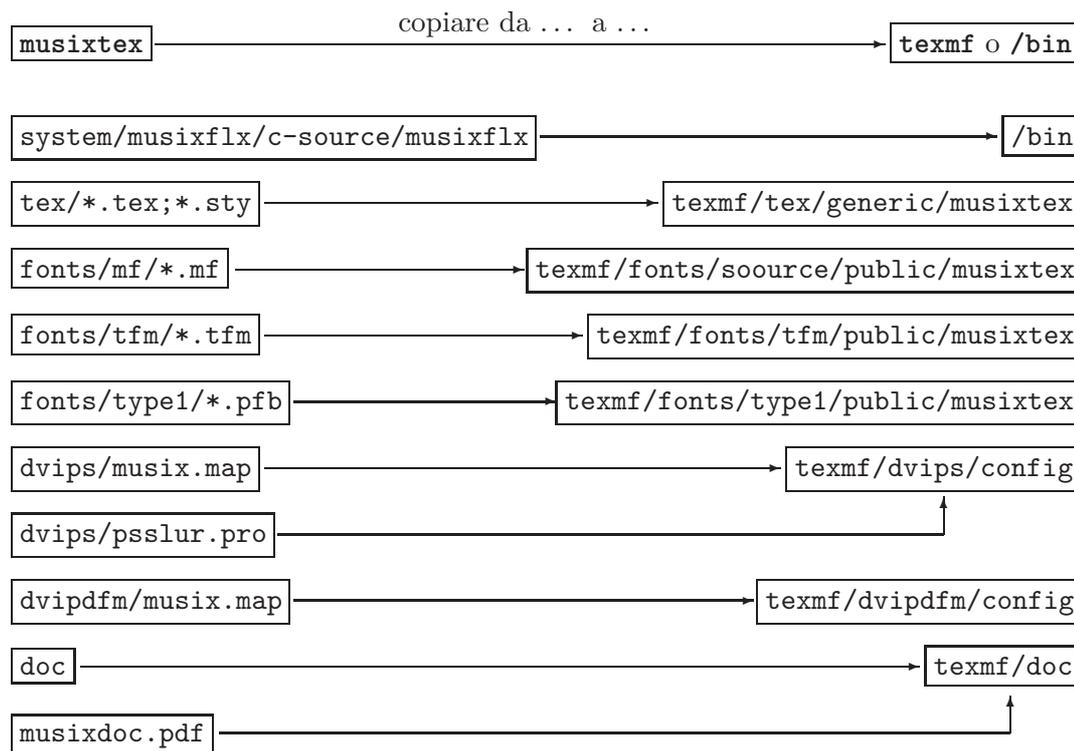


TABELLA 1.5 – Installazione di MusiXTeX (Linux)

```
cp scor2prt.for scor2prt.f
```

Poi aprire con un editor il file `pmxab.f`, individuare le linee contenenti il comando `getarg` come quelle che seguono

```

c    call getarg(1,jobname,idum) ! May need to replace this w/ next line
c    call getarg(1,jobname)

```

e modificarle in questo modo:

```

c    call getarg(1,jobname,idum) ! May need to replace this w/ next line
c    call getarg(1,jobname)

```

poi salvare e uscire. Fare la stessa operazione con `scor2prt.f`.

Ora si devono trasformare i sorgenti dei due programmi da Fortran a C:

```
f2c pmxab.f -Nx400 -Nn802
```

```
f2c scor2prt.f -Nx400 -Nn802
```

Compilare i programmi:

```
gcc pmxab.c -lf2c -lm -o pmxab
```

```
gcc scor2prt.c -lf2c -lm -o scor2prt
```

Si possono anche compilare direttamente i sorgenti Fortran¹² senza la conversione in C, in questo modo:

```
g77 pmxab.f -o pmxab
```

```
g77 scor2prt.f -o scor2prt
```

Ora siamo pronti per copiare i file nell'albero T_EX come indicato nella figura 1.6.

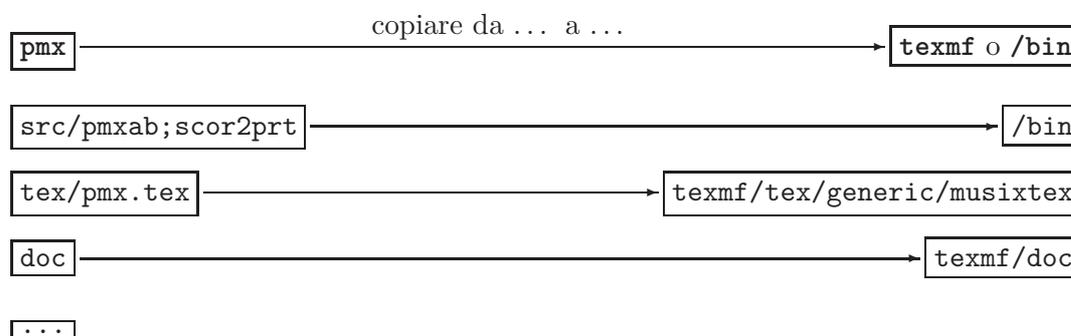


TABELLA 1.6 – Installazione di **PMX** (Linux)

Rigenerare infine il database con `texthash`.

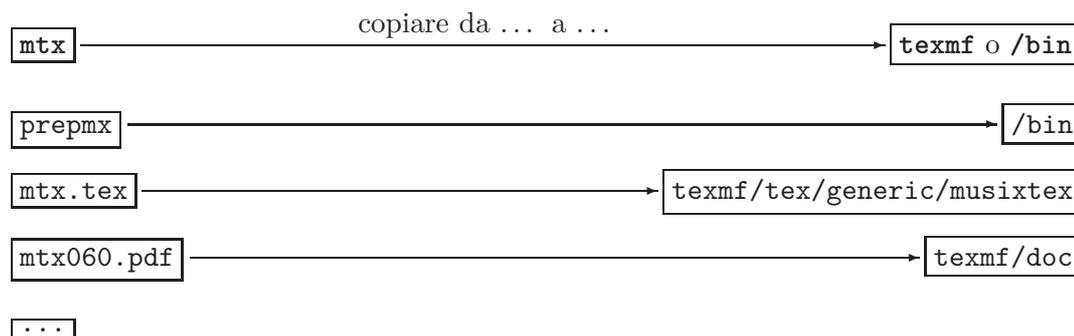
M-Tx

Dal **Werner Icking Music Archive** (<http://icking-music-archive.org>) scaricare l'archivio `mtx060d.zip` e decomprimerlo in una directory temporanea (es.: `mtx`). Nella stessa directory scaricare anche `mtx060.pdf` (la documentazione). Anche in questo caso dobbiamo compilare il programma. Lo si può fare semplicemente ponendosi all'interno della directory temporanea e digitando il comando `make`¹³, oppure con il comando `fpc -B -So prepmx`. Fatta la compilazione si può procedere alla copiatura dei file nell'albero T_EX come indicato nella figura 1.7.

Infine: `texthash`.

¹²Deve però essere installato il compilatore `g77`.

¹³Si può a questo punto anche dare il comando `make install` per l'installazione automatica, ma sono necessari i privilegi di root, oppure è necessario modificare le impostazioni previste per la compilazione.

TABELLA 1.7 – Installazione di **M-Tx** (Linux)

Pacchetto per le legature PostScript tipo M

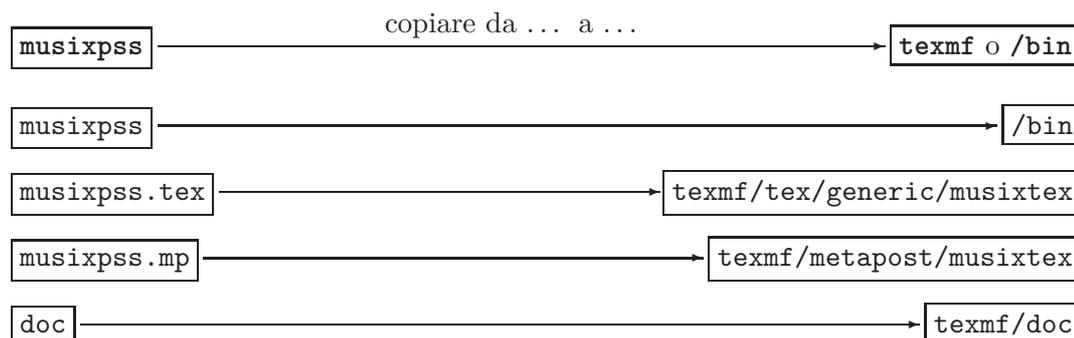
Scaricare (come indicato a pagina 1.2.1) e decomprimere l'archivio in una directory temporanea (es.: `musixpss`). Modificare le linee 44 – 45 di `musixpss.c` in modo che risultino così:

```
/* #define PATH_DELIMITER '\\\ ' for MS-DOS, MS-Windows */
#define PATH_DELIMITER '/' /* for UNIX and clone */
```

Poi compilare con il comando

```
gcc musixpss.c -o musixpss
```

Copiare come indicato nella figura 1.8.

TABELLA 1.8 – Installazione di **Musixpss** (Linux)

Rigenerare il database con `texhash`.

1.3 Uso dei programmi

1.3.1 T_EX

Alla base di tutto il sistema c'è, come si è detto, il prestigioso programma di tipografia elettronica T_EX. Il programma legge un semplice file di caratteri leggibili dall'uomo, contenente il testo da stampare, completo dei comandi necessari per definire gli tutti aspetti tipografici della pagina, salvato normalmente con estensione `.tex` e genera un file con estensione `.dvi`¹⁴ (*device-independent*), non più leggibile come il precedente dall'uomo, ma contenente tutte le informazioni per la visualizzazione grafica e la stampa. Il file `.dvi` può essere letto da un apposito programma compreso nella distribuzione T_EX¹⁵, che lo visualizza sullo schermo ed è in grado di stamparlo, oppure può essere trasformato in un file `.ps` (postscript) o `.pdf` (Portable Document Format), per una stampa di maggior qualità. Per produrre un file `.dvi` stampabile da un file sorgente `.tex` è sufficiente lanciare da riga di comando

```
tex nomefile
```

È possibile poi produrre il file `.ps` e quello `.pdf` rispettivamente con i seguenti comandi:

```
dvips nomefile
ps2pdf nomefile
```

Per produrre direttamente un file `.pdf`, senza passare dai formati `.dvi` e `.ps`, il comando è:

```
pdftex nomefile
```

1.3.2 MusiX_TE_X

La giustificazione di una linea di musica presenta problemi diversi rispetto a quella di una linea di testo. Per questo è stato studiato dagli autori di MusiX_TE_X un sistema in tre passi anziché in uno. Dopo la prima compilazione con T_EX (primo passo) le informazioni per il calcolo della giustificazione delle linee vengono scritte in un file esterno (`nomefile.mx1`). A questo punto si lancia `musixflx`, un programma accessorio di MusiX_TE_X, che produce un altro file esterno, `nomefile.mx2` contenente i dati corretti per la giustificazione. Si processa infine un'altra volta `nomefile.tex` e si ottiene il file `nomefile.dvi` definitivo. Attenzione: dopo ogni

¹⁴In realtà sono generati sempre almeno altri due file, di cui non è necessario interessarsi ora: un file `.log` contenente i messaggi che appaiono anche sullo schermo durante la compilazione, utili per la correzione di eventuali errori e un file `.aux`, contenenti i riferimenti incrociati nel testo.

¹⁵In Mik_TE_X Yap; nei sistemi Linux `xdvi`, `kdvi` o `evince`.

compilazione i file `.mx1` e `.mx2` vanno cancellati, altrimenti forniscono dati errati durante le successive elaborazioni della partitura. Riassumendo, ecco dunque la procedura.

1. `tex nomefile`
2. `musixflx nomefile`
3. `tex nomefile`

Ecco tutti i passaggi fino alla stampa:

1. Creazione del file `nomefile.tex` con un editor di testi
2. `tex nomefile`¹⁶ (creazione di `nomefile.dvi` e `nomefile.mx1`)
3. `musixflx nomefile` (creazione di `nomefile.mx2`)
4. `yap nomefile` (visualizzazione sullo schermo della partitura). In Linux si può usare `xdvi`, `evince` o `kdvi`. Il funzionamento di ciascuno è analogo; per esempio, nel primo caso il comando è `xdvi nomefile`.
5. `dvips nomefile` (creazione del file postscript `nomefile.ps`)¹⁷
6. Visualizzazione e eventuale stampa in postscript. Aprire **GhostView** e caricare il file `nomefile.ps`¹⁸. Nei sistemi Linux sono disponibili diversi visualizzatori postscript (`kghostview`, `evince` o `gv` sono i più noti). In quest'ultimo caso, per esempio, da linea di comando, digitare `gv nomefile`.

I passi 5 e 6 sono obbligatori solo se si usano le legature postscript tipo K. Per produrre un file `.pdf` si hanno a disposizione 3 possibilità:

1. Convertire il file `nomefile.ps` ottenuto nel modo appena descritto in pdf con il comando `ps2pdf nomefile.ps`. Attenzione: questo è il solo metodo disponibile se si usa il pacchetto K per le legature.
2. Compilare con `pdftex`. I passaggi sopra descritti, se non sono impiegate legature postscript, si riducono a tre:
 - (a) `pdftex nomefile`
 - (b) `musixflx nomefile`
 - (c) `pdftex nomefile`
3. Convertire il file `nomefile.dvi` con `dvipdfm` usando il comando:


```
dvipdfm nomefile
```

¹⁹

¹⁶L'estensione può essere omessa.

¹⁷**Dvips** è un programma presente in ogni distribuzione di T_EX.

¹⁸Dovrebbero essere sufficienti anche solo due click sull'icona del file.

¹⁹Procedura anche in questo caso non disponibile con le legature postscript tipo K.

1.3.3 PMX e M-Tx

Normalmente per la creazione della partitura si usano i preprocessori descritti in questa guida. Ciò impone che siano fatti uno o due passaggi in più per ottenere il file di stampa. Se si usa **M-Tx** la codifica della partitura andrà salvata in un file di estensione `.mtx`. Il programma `prepmx` produce un file `.pmx` che a sua volta andrà compilato con `pmxab` per ottenere il file `.tex`. Se la partitura è codificata in **PMX** sarà ovviamente necessario solo il secondo passaggio. Ecco l'elenco completo dei passaggi necessari in questi due casi.

PMX

1. `pmxab nomefile`
2. `tex nomefile`
3. `musixflx nomefile`
4. `tex nomefile`
5. `dvips nomefile`
6. `ps2pdf nomefile`

M-Tx

1. `prepmx nomefile`
2. `pmxab nomefile`
3. `tex nomefile`
4. `musixflx nomefile`
5. `tex nomefile`
6. `dvips nomefile`
7. `ps2pdf nomefile`

Con `pdftex`, ma senza legature postscript, la compilazione è invece:

PMX

1. `pmxab nomefile`
2. `pdftex nomefile`
3. `musixflx nomefile`
4. `pdftex nomefile`

M-Tx

1. `prepmx nomefile`
2. `pmxab nomefile`
3. `pdftex nomefile`
4. `musixflx nomefile`
5. `pdftex nomefile`

1.3.4 Utilizzo con le legature postscript tipo M

Il pacchetto postscript tipo M richiede **metapost**, un sistema di disegno postscript di norma presente nelle principali distribuzioni di $\text{T}_{\text{E}}\text{X}$. Le figure prodotte da questo programma possono essere richiamate da $\text{T}_{\text{E}}\text{X}$ e integrate nei file di uscita `.ps` e `.pdf` attraverso `pdftex`, `dvips` o `dvipdfm`. **musixpss**, si serve di **metapost** per disegnare tutte le legature presenti in una partitura e $\text{T}_{\text{E}}\text{X}$ le include poi nel file di stampa. Con questo metodo tutte le procedure per la produzione di file di stampa sono disponibili. Sono però necessari più passi per la compilazione, inoltre vengono generati numerosi file intermedi. Il programma permette 3 modi operativi diversi:

- ‘general’ (compatibile con tutti i formati di stampa),
- ‘dvips’ (ottimizzato per `dvips`) e

- ‘pdftex’ (ottimizzato per pdftex).

È inoltre possibile la selezione automatica fra due delle tre precedenti opzioni alternative:

- `\psslurmode{auto-pdftex-general}`
- `\psslurmode{auto-pdftex-dvips}`

Nel primo caso il programma riconosce se la compilazione viene fatta con \TeX o con `pdftex` e sceglie di conseguenza la configurazione ottimale; nel secondo caso la scelta avviene automaticamente fra `pdftex` e `dvips/dvipdfm`. Poiché per stampe accurate è necessario produrre file `.ps` o `.pdf`, illustreremo qui solo la compilazione nel modo `\psslurmode{auto-pdftex-dvips}`. Per ulteriori dettagli si veda la documentazione allegata al programma.

Per compilare la partitura con questo sistema è necessario invocare il pacchetto nel preambolo, dopo aver richiamato **MusiX \TeX** , nel seguente modo:

```
\input musixtex
\input musixpss
\relax
\psslurmode{auto-pdftex-dvips}%
```

Se si impiega il codice **PMX** o **M-Tx** si dovranno usare i comandi \TeX in linea come descritto nel paragrafo 2.3.27. Ecco i comandi:

PMX

```
\\input musixpss\relax\
\\psslurmode{auto-pdftex-dvips}%\
```

M-Tx

```
%%\\input musixpss\relax\
%%\\psslurmode{auto-pdftex-dvips}%\
```

La compilazione va fatta in modo diverso a seconda che si voglia produrre un file `.ps` con `dvips/dvipdfm` o un file `.pdf` con `pdftex`; ecco i passi:

dvips/dvipdfm

1. `tex nomefile`
2. `musixflx nomefile`
3. `tex nomefile`
4. `musixpss nomefile`
5. `mpost nomefile`
6. `dvips (o dvipdfm) nomefile`

pdftex

1. `pdftex nomefile`
2. `musixflx nomefile`
3. `pdftex nomefile`
4. `musixpss nomefile`
5. `mpost nomefile`
6. `pdftex nomefile`

Alla fine di ogni compilazione è consigliabile eliminare i file temporanei prodotti dal programma con il comando `musixpss -clean nomefile`.

1.3.5 Compilazione automatica

I diversi passi della compilazione di un file sorgente possono essere resi automatici riunendo in sequenza i comandi necessari in uno script (per Linux) o in un file batch (per Windows). In questo modo sarà possibile effettuare l'intero processo con un semplice comando.

Windows

Ecco un esempio di file batch per Windows per la compilazione da codice **M-Tx** a **.ps**, via **dvips**²⁰. Per far ciò basta aprire un editor di testo (ad esempio **edit**), scrivere le linee che seguono, poi salvare il file con il nome **mtx2ps.bat** (qualsiasi nome va bene, ma l'estensione deve essere sempre **.bat**.) nella directory **mieibat** precedentemente creata (o in un'altra inclusa nel path).

```
% mtx2ps.bat
echo off
prepmx %1
if errorlevel 1 then goto exit
echo off
pmxab %1
if errorlevel 1 then goto exit
echo off
del *.mx?
tex %1
if errorlevel 1 then goto exit
musixflx %1
tex %1
if errorlevel 1 then goto exit
dvips %1
```

Ogni linea che inizia con **%** è una linea di commento e non verrà utilizzata ai fini della compilazione. Il segno **%1** significa che il programma prenderà il nome del file dalla linea di comando. Per compilare una partitura codificata in **M-Tx** basterà quindi aprire una finestra dos nella posizione in cui si trova il file da compilare (diciamo **nomefile.mtx**) e digitare

```
mtx2ps nomefile
```

Tutti i passaggi necessari saranno compiuti automaticamente e, se non vi sono errori nel file sorgente, si otterrà alla fine della compilazione **nomefile.ps** contenente la partitura pronta per essere visualizzata sullo schermo o stampata.

Sarà bene creare per lo stesso scopo altri due batch file (propongo di chiamarli **pmx2ps.bat** e **mtx2ps.bat**²¹) che serviranno alla compilazione a partire rispettivamente da un codice **PMX** e **MusiX \TeX** . Il loro contenuto sarà uguale al

²⁰Questa procedura è adatta a partiture contenenti legature bitmap o postscript tipo K.

²¹Assicuratevi però che tali nomi non corrispondano a quelli di altri programmi presenti sul vostro sistema, altrimenti potrebbero verificarsi dei conflitti.

precedente, ma escluderà le parti non necessarie. In `pmx2ps.bat` non dovranno dunque essere incluse le linee 2–5, mentre la prima linea sarà mantenuta ma con l’indicazione esatta del nome del file. In `mtex2ps.bat`, analogamente, si dovrà correggere il nome del file nella prima linea e si dovranno eliminare le linee 2–8.

Ecco ancora il batch `mtx2ps.bat` per la compilazione da codice **M-Tx** a `.ps` con legature postscript tipo M.

```
% mtx2ps.bat
echo off
prepmx %1
if errorlevel 1 then goto exit
echo off
pmxab %1
if errorlevel 1 then goto exit
echo off
del *.mx?
tex %1
if errorlevel 1 then goto exit
musixflx %1
tex %1
if errorlevel 1 then goto exit
musixpss %1
if errorlevel 1 then goto exit
mpost %1
if errorlevel 1 then goto exit
dvips %1
if errorlevel 1 then goto exit
musixpss -clean %1
```

Ed ecco `mtx2pdf.bat` per la compilazione da codice **M-Tx** a `.pdf` con legature postscript tipo M.

```
% mtx2pdf.bat
echo off
prepmx %1
if errorlevel 1 then goto exit
echo off
pmxab %1
if errorlevel 1 then goto exit
echo off
del *.mx?
pdftex %1
if errorlevel 1 then goto exit
musixflx %1
```

```

pdftex %1
if errorlevel 1 then goto exit
musixpss %1
if errorlevel 1 then goto exit
mpost %1
if errorlevel 1 then goto exit
pdftex %1
if errorlevel 1 then goto exit
musixpss -clean %1

```

Effettuando le modifiche già viste sopra²² sarà possibile ottenere anche `pmxm2ps`, `texm2ps`, `pmxm2pdf` e `texm2pdf` per compilazioni a partire da partiture codificate in `.pmx` e in `.tex`.

Linux

Uno script per la shell di Linux (ma sarebbe più corretto dire Unix), che in genere è `bash`, ha di norma estensione `.sh`. Al suo interno le righe di commento iniziano con il carattere `#`. Per eseguire il comando si deve aprire un terminale (shell) nella posizione in cui si trova il codice sorgente da compilare e (poniamo il caso di usare `mtx2ps.sh`) digitare

```
sh mtx2ps.sh nomefile
```

Presento di seguito gli equivalenti dei file batch già visti per Windows.

```
mtx2ps.sh
```

```

#!/bin/bash
# mtx2ps.sh
if [ -e $1.mtx ]; then
prepmx $1.mtx
else
  echo Errore: $1.mtx non esiste
fi
if [ -e $1.pmx ]; then
  pmxab $1.pmx
else
  echo Errore: $1.pmx non esiste
fi
rm -f $1.dvi $1.mx?
if [ -e $1.tex ]; then
  tex $1.tex
  musixflx $1.tex

```

²²Esclusione delle linee 2-5 o 2-8 e modifica del nome del file in linea 1.

```

tex $1.tex
else
  echo Errore: $1.tex non esiste
fi
if [ -e $1.dvi ]; then
  dvips $1.dvi -o $1.ps
else
  echo Errore: $1.dvi non esiste
fi

```

```

mtxm2ps.sh

```

```

#!/bin/bash
# mtxm2ps.sh
if [ -e $1.mtx ]; then
  prepmx $1.mtx
else
  echo Errore: $1.mtx non esiste
fi
if [ -e $1.pmx ]; then
  pmxab $1.pmx
else
  echo Errore: $1.pmx non esiste
fi
rm -f $1.dvi $1.mx?
if [ -e $1.tex ]; then
  tex $1.tex
  musixflx $1.tex
  tex $1.tex
  musixpss $1.slu
  mpost $1.mp
else
  echo Errore: $1.tex non esiste
fi
if [ -e $1.dvi ]; then
  dvips $1.dvi -o $1.ps
  musixpss -clean $1.slu
else
  echo Errore: $1.dvi non esiste
exit
fi

```

```

mtxm2pdf.sh

```

```

#!/bin/bash

```

```

# mtxm2pdf.sh
if [ -e $1.mtx ]; then
  prepmx $1.mtx
else
  echo Errore: $1.mtx non esiste
fi
if [ -e $1.pmx ]; then
  pmxab $1.pmx
else
  echo Errore: $1.pmx non esiste
fi
rm -f $1.dvi $1.mx?
if [ -e $1.tex ]; then
  pdftex $1.tex
  musixflx $1.tex
  pdftex $1.tex
  musixpss $1.slu
  mpost $1.mp
  pdftex $1.tex
  musixpss -clean $1.slu
else
  echo Errore: $1.tex non esiste
fi

```

Per script di compilazione a partire da codice **PMX** o da codice **MusiX_TE_X** si devono omettere rispettivamente le linee 3 – 7 e 3 – 12 di ognuno degli esempi appena proposti e si deve correggere nella linea 2 di ciascuno il nome dello script.

1.4 Gli autori dei programmi

- **MusiX_TE_X**. È l'erede di MusiC_TE_X di *Daniel Taupin* (1936 – 2003). Vi hanno collaborato, oltre allo stesso Taupin, *Ross Mitchell* (CSIRO Division of Atmospheric Research, Private Bag No.1, Mordialloc, Victoria 3195, Australia) e *Andreas Egler* (Ruhr–Uni–Bochum, Ursulastr. 32, D-44793 Bochum). È attualmente mantenuto e periodicamente aggiornato da Don Simons, Jean-Pierre Coulon ed Hiroaki Morimoto. Gli aggiornamenti si trovano nel <http://icking-music-archive.org>.
- **PMX** *Don Simons* (Dr. Don's PC and Harpsicord Emporium, Redondo Beach, California, USA, e-mail: *vedi* indirizzo n. 4 di figura 1.2, pag. 25).
- **M-Tx** *Dirk Laurie* (e-mail: *vedi* indirizzo n. 3 di figura 1.2, pag. 25)

- **Musixlyr** (pacchetto ora incluso nella distribuzione principale di **MusiX \TeX**) *Rainer Dunker* (e-mail: *vedi* indirizzo n. 7 di figura 1.2, pag. 25)
- **Pacchetto per le legature postscript tipo K** (ora incluso nella distribuzione principale di **MusiX \TeX**) *Stanislav Kneift* (e-mail: *vedi* indirizzo n. 9 di figura 1.2, pag. 25)
- **Pacchetto per le legature postscript tipo M** *Hiroaki Morimoto* (e-mail: *vedi* indirizzo n. 2 di figura 1.2, pag. 25)

1.5 Il Werner Icking Music Archive

Werner Icking (25/6/1943–8/2/2001) fu il fondatore del **GMD Music Archive**, (<http://www.gmd.de/Misc/Music/>) contenente un archivio di partiture, i programmi del gruppo **MusiX \TeX** , una lista di discussione ed altre risorse musicali gratuite. Il suo contributo allo sviluppo e soprattutto alla diffusione di questi software è stato impareggiabile. Le sue edizioni musicali (in particolare le edizioni complete ‘urtext’ delle opere per violino e per violoncello solo di Bach) sono esempi di raffinatezza tecnica, sensibilità artistica e fedeltà filologica. Chiunque, fosse egli un esperto programmatore o un principiante assoluto, poteva trovare in lui un valido e disinteressato aiuto, chiuso immancabilmente dal motto *hope this hilft*. Attorno al sito si è raccolta una non grande, ma costante comunità di appassionati di tutto il mondo, che, pur non avendo nella maggior parte dei casi mai visto Werner di persona, era abituata ad avere amichevoli e frequenti rapporti epistolari con lui. Werner morì improvvisamente l’8 febbraio 2001 mentre, come faceva sempre, tornava a casa dal lavoro in bicicletta. Quel giorno aveva curato il sito e risposto ad alcuni messaggi della lista di discussione, di cui, oltre ad essere il fondatore, era anche il principale animatore. La comunità dei frequentatori del sito ne fu sconvolta e decise di proseguire l’opera di Werner. Un nuovo sito, dedicato alla sua memoria, è stato dunque fondato, il **Werner Icking Music Archive** (<http://icking-music-archive.org>), per continuare l’opera intrapresa da lui e da tutta la comunità. Ecco come è organizzato.

- **Archivio di partiture gratuite** vi si trovano centinaia di partiture gratuite in formato Acrobat PDF, spesso con i codici sorgente. È mantenuto da *Christian Mondrup* (e-mail *vedi* indirizzo n. 8 di figura 1.2, pag. 25), che è anche il principale animatore del sito.
- **Sezione software** Distribuisce tutti i programmi musicali del gruppo **MusiX \TeX** , i manuali e le utility accessorie. È mantenuto da *Don Simons* (e-mail: *vedi* indirizzo n. 4 di figura 1.2, pag. 25).
- **Lista di discussione MusiX \TeX** È una risorsa preziosa per chi utilizza i programmi. Si occupa sia di problemi tecnici, che di problemi musicali, di

carattere storico o estetico. È in inglese ed è curata da *Maurizio Codogno*. La pagina di accesso e di registrazione si trova all'indirizzo <http://mailman.nfit.au.dk/mailman/listinfo/icking-music-archive.org-tex-music>.

- **Pagina di link sulle risorse musicali informatiche** Contiene indicazioni su motori di ricerca musicali, altri archivi di partiture on line, cataloghi di opere e altre risorse musicali. È mantenuta da *Jean-Pierre Coulon* (e-mail: vedi indirizzo n. 1 di figura 1.2, pag. 25).

1. coulon@obs-nice.fr	6. MMolekwa@rrs.co.za
2. CQX05646@nifty.ne.jp	7. rainer.dunker@web.de
3. dlaurie@na-net.ornl.gov	8. reccmo@daimi.au.dk
4. dsimons@roadrunner.com	9. standa@hiero.cz
5. luicatal@interfree.it	

FIGURA 1.2 – e-mail citate nel testo

Capitolo 2

Manuale d'uso di PMX

2.1 Concetti preliminari

La struttura di un codice **PMX** è composta da due parti: il *preambolo* e il *corpo*. Nel *preambolo* sono specificati i caratteri generali della partitura, sia musicali (il numero degli strumenti, il metro, gli accidenti ecc.) che tipografici (il numero delle pagine, il numero dei sistemi¹ per pagina, la misura dei rientri ecc.). Nel *corpo* si trova la musica vera e propria. Le righe del codice che iniziano con il segno % sono di commento e vengono dunque ignorate dal programma durante la trascrizione della partitura. Prendiamo ad esempio le battute iniziali del *Quartetto Op. 76, n. 2* di Haydn. La musica di figura 2.1 è stata ottenuta dal seguente codice **PMX**.

```
% PREAMBOLO
4 4 4 4 0 6 0 -1
1 1 20 .07
Vc
Va
Vl II
Vl I
batt
./
% CORPO
Ab
It92ivcvavlvl
w170m
% Batt 1
h
Allegro
d8-- Df o. d+ o. d o. d o. r d o. d o. d o. /
f8- Df o. f o. f o. f o. r f o. f o. f o. /
r8 a Df o. a o. a o. r a o. a o. a o. /
```

¹Il termine più corretto in italiano è *accollatura*, tuttavia l'italianizzazione dell'inglese *system* è ormai comunemente usata e, per comodità, verrà impiegata anche in questo manuale.

a2+ Df d- /

% Batt 2

r8 d o. d o. d o. r e o. e o. e o. /
 r8 g o. g o. g o. r g o. g o. g o. /
 r8 b o. b o. b o. r c s o. c o. c o. /
 e2 a- /

% Batt 3-4

f8 s e f c s s d4 a | b2 s a4 s o. r Rb /
 a2 t a4 t .cs- | d8 s e f d s e4 o. r /
 d8 s c s d e s e s d c s a1+ s g s | f s g f e s d8 o. d o. cs4 r /
 d8 s c s d e s g s f e s a | d4- zd+ cn1 s b a g s a4 o. r /

Allegro

FIGURA 2.1 – F. J. Haydn, *Quartetto Op. 76 n. 2*, batt. 1-4

2.2 Preambolo

Il *preambolo* è composto da due righe di numeri, seguite da poche altre righe, contenenti per lo più caratteri alfabetici e ha lo scopo di fornire informazioni che valgono per l'intera partitura.

La prima riga di numeri contiene informazioni di carattere musicale, la seconda di carattere tipografico. Va detto che questi numeri avrebbero potuto essere scritti indifferentemente sulla stessa linea o su linee diverse. Nella lingua di **PMX**, come in quella di **T_EX**, l'‘a capo’ è uguale allo spazio. Perciò le seguenti tre forme sono del tutto analoghe:

4 4 4 4 0 6 0 -1
 1 1 20 .085

4 4 4 4 0 6 0 -1 1 1 20 .085

4
4
4
4
0
6
0
-1
1
1
20
.085

I primi due numeri servono a definire il numero di pentagrammi² e il numero di strumenti. Normalmente si tratta di due numeri uguali, ma per strumenti che richiedono due pentagrammi, come il pianoforte, il numero di pentagrammi deve essere superiore al numero di strumenti. In questo caso **PMX** predispone il rigo di pianoforte (due pentagrammi uniti da una graffa) per lo strumento più in basso nel sistema.

Questo sistema può essere impiegato anche in altri casi usando il segno meno davanti alla seconda cifra (numero di strumenti). In questo caso le cifre immediatamente seguenti indicheranno il numero di pentagrammi per ciascuno strumento (dal basso verso l'alto). Per esempio per impostare un sistema adatto alla musica d'organo, recante i primi due pentagrammi uniti dalla graffa per i due manuali ed il terzo per il pedale si dovrà scrivere 3 -2 1 2. Se invece volessimo predisporre un sistema per tre coppie di violini dovremmo scrivere 4 -3 2 2 2 ed otterremmo sei pentagrammi riuniti a coppie da graffe. Come si vede in questo caso vi sono ben 5 cifre al posto delle consuete due.

I quattro numeri che seguono servono a definire il metro. I primi due (numeratore e denominatore) indicano il metro *logico*, cioè quello in base al quale **PMX** calcola la lunghezza della battuta, mentre gli altri due stabiliscono il metro che verrà stampato nella partitura. Può infatti essere necessario scrivere un valore diverso rispetto a quello logico (come ad esempio capita nel caso del tempo 'a cappella') o non stampare affatto il metro. Per esempio 4 4 4 4 indica una composizione in quattro quarti in cui il metro è indicato mediante la medesima frazione, 2 2 0 5 indica il tempo tagliato e 4 4 0 6 il tempo 'a cappella'. Quando il terzo numero, cioè il numeratore stampato è uguale a 0, è possibile immettere come valore del denominatore stampato 0, 1, 2, 3, 4, 5, 6 o 7. La tabella 2.1 spiega il significato di queste combinazioni, mentre l'effetto è mostrato nella figura 2.2.

²Nel manuale è indicato come numero di voci, ma mi pare che ciò produca qualche confusione con il numero di voci che possono essere scritte sullo stesso pentagramma.

0	non verrà stampato alcun metro (<i>a</i>)
1 2 3 4	verrà stampato un solo numero centrato verticalmente (<i>b c d e</i>)
5	verrà stampato il tempo tagliato (alla breve) (<i>f</i>)
6	verrà stampato il tempo 'a cappella' (<i>g</i>)
7	verrà stampato un 3 sbarrato verticalmente (<i>h</i>)

TABELLA 2.1 – Casi particolari di indicazione del metro



FIGURA 2.2 – Casi particolari di indicazione del metro

I valori *logici* introdotti devono corrispondere a quelli previsti come durate delle note (che vedremo fra poco). Per questo è necessario fare attenzione a due casi particolari. Per metri $n/1$, si deve usare 0 come **denominatore logico** e 1 come **denominatore stampato**, poiché 0 indica nella lingua di **PMX** la semibreve; ad esempio per ottenere $3/1$ si deve scrivere 3 0 3 1. Per i metri $n/16$ si deve usare 1 come **denominatore logico** (1 = semicroma) e 16 come **denominatore stampato**; ad esempio per ottenere $24/16$ si deve scrivere 24 1 24 16.

Se la partitura comincia con un levare, il settimo numero della lista serve ad indicarlo³. Si tratta di un numero intero o decimale che rappresenta la frazione di battuta corrispondente alla lunghezza del levare. Nel $4/4$, per esempio, la semiminima corrisponde a $1/4$, perciò se il levare è del valore di una semiminima scriveremo 1, se è una croma 0.5, come si vede nell'esempio di figura 2.3. Allo stesso modo, se il metro è di $3/8$, il levare di croma corrisponde a 1, quello di semicroma a 0.5 e così via.

FIGURA 2.3 – Valori del levare in $4/4$

L'ottavo numero indica l'armatura. Se il numero è positivo (senza il segno) saranno introdotti dei diesis, se è negativo dei bemolli.

I restanti quattro numeri definiscono parametri tipografici. Il primo stabilisce il numero delle pagine, il secondo il numero totale dei sistemi o, se il numero delle pagine è uguale a 0, il numero di battute per sistema (è utile quando si comincia

³Questa funzione vale solo se il levare è all'inizio della partitura. Se si trova all'inizio di un movimento a metà del codice **PMX** si deve introdurre un cambio di metro invisibile (*cfr* paragrafo 2.3.23).

ad introdurre la musica battuta per battuta). Va poi indicata la grandezza del pentagramma (16, 20, 24 o 29 punti) e il rientro della prima riga di musica all'inizio di ciascun movimento.

La tabella 2.2 contiene il riepilogo dei 12 numeri del preambolo.

Valori musicali (numeri 1–8)

Num.	1	2	3	4	5	6	7	8
Funz.	num. di pentagr.	num. di strum.	numerat. logico	denomin. logico	numerat. stampato	denomin. stampato	levare	armatura

Valori tipografici (numeri 9–12)

Numero	9	10	11	12
Funzione	numero di pagine (o 0)	numero di sistemi (o batt. per sist.)	grandezza del pentagramma	rientro

TABELLA 2.2 – Riepilogo dei valori del *preambolo*

La tabella 2.3 illustra il significato della lista dei numeri del preambolo dell'esempio di figura 2.1

4	4 pentagrammi	1	una pagina
4	4 strumenti	1	un sistema
4	numeratore logico	20	grandezza del pentagramma
4	denominatore logico	.085	rientro
0	numeratore stampato		
6	denominatore stampato		
0	nessun levare		
-1	un bemolle in chiave		

TABELLA 2.3 – I valori numerici del preambolo di figura 2.1

Dopo la lista di numeri, disposti su righe successive, vanno introdotti i nomi degli strumenti⁴. Il numero delle righe è quello indicato dal secondo numero della lista. Nel caso che non si desideri scrivere alcun numero si devono lasciare le righe vuote.

La riga successiva è riservata all'indicazione delle chiavi da inserire in ciascun pentagramma, dal basso verso l'alto. I simboli usati sono **t**, **s**, **m**, **a**, **n**, **r**, **b**, **f** e significano, rispettivamente violino (inglese: *treble*), soprano, mezzo soprano, contralto, tenore, baritono, basso, e chiave di violino francese⁵. Al posto delle lettere

⁴ Vedi ad esempio le righe 3–6 del codice riprodotto a pagina 27.

⁵ Nel codice di pagina 27 le chiavi che appaiono alla riga 7 sono: basso, contralto, violino e violino (**batt**).

è possibile usare i numeri per indicare le chiavi, secondo il sistema di codifica valido per **MusiX_{TEX}**, cioè: 0 (violino), 1 (soprano), 2 (mezzo soprano), 3 (contralto), 4 (tenore), 5 (baritono), 6 (basso), 7 (violino francese). I due sistemi alternativi di codifica sono riassunti in figura 2.4.



FIGURA 2.4 – Chiavi

L'ultima riga del preambolo permette di specificare la directory in cui si vuole che **PMX** scriva il codice **TEX**. `./` indica la stessa directory in cui si trova il file **PMX**, ed è l'opzione che si usa di solito.

2.3 Corpo

Il corpo del codice **PMX** contiene la musica vera e propria ed è diviso in *blocchi*.

Ciascun blocco può contenere da una a quindici battute musicali. Non è necessario separare le battute con l'indicazione della stanghetta, ma è buona norma farlo usando il simbolo `|`, per chiarezza e per permettere un controllo più agevole dei valori delle note nella battuta. Ogni blocco è composto da diverse righe. Ciascuna riga contiene una voce musicale. Le voci sono disposte dalla più bassa alla più alta, quindi la voce più in alto nel codice **PMX** corrisponde a quella più in basso nella partitura e viceversa. Normalmente (è il caso ad esempio degli strumenti omofonici) un pentagramma contiene una sola voce. All'interno di ciascun blocco, ogni pentagramma si conclude col simbolo `/`. Se un pentagramma contiene più di una voce la prima delle due voci (la più in alto nel codice e la più in basso nella partitura) si conclude con il simbolo `//` e la successiva, quella che conclude il pentagramma, con il simbolo `/`. Le due battute del *Pelléas et Melisande* di Debussy di figura 2.5 ce ne mostrano un esempio. Il pentagramma inferiore (tromba con sordina) contiene una voce, mentre il pentagramma superiore (flauti) ne contiene due, per un totale di tre voci. In questo caso ogni blocco di **PMX** dovrà contenere tre righe di musica, una per ciascuna voce e, per indicare che nel pentagramma superiore va inserita una seconda voce, la voce più grave si conclude col simbolo `//`. Ecco il codice **PMX** dell'esempio di figura 2.5.

```
2 2 12 8 12 8 0 6
%
% npag nsist pent rientri
1 1 20 .07
Tromba
Flauti
tt
./
```

```

Ab
It64itrfl
w167m
% Batt 1
h-4
Lento
cd4 D"con sord."+17 s dd fd dd s /
r8+0 fr s-6+1 e s-5 t ed4 t r8+0 br+ s-6+1 a s-5 r+0 gr s-6+1 f s-5 //
rb8 g4+ t gd t rb8 c4 rb8 a4 /

% Batt 2
cd4 s dd fd ad s Rb /
r8+0 fr s-6+1 e s-5 t ed4 t r8+0 br+ s-6+1 a s-5 t ad4 t //
rb8 g4 t gd t rb8 c4 t cd t /

```

PMX permette di variare il numero delle voci ad ogni blocco. Ciò è molto utile quando in un pentagramma vi sono a volte due voci e a volte una sola voce. Se una delle linee del codice del blocco si conclude con //, **PMX** accetta i simboli necessari per una seconda voce, che si concluda con /; se non compare //, ma solo /, allora **PMX** permette l'introduzione di una sola voce in quel pentagramma.

Ciascuna linea del codice, come si vede dagli esempi sin qui mostrati, contiene una serie di simboli che descrivono i segni musicali che saranno stampati sulla partitura. Per le legature, l'indicazione dei gruppi ritmici, le forcelle del crescendo e del diminuendo e altri simili segni che possono estendersi in lunghezza o in altezza, è necessario introdurre un simbolo di inizio e uno di fine. I caratteri alfanumerici che servono a descrivere uno stesso segno musicale vanno introdotti uno dopo l'altro, senza spazi. Ciascun simbolo è invece separato da quello successivo mediante uno spazio.

Lento

Flauti

Tromba *con sord.*

FIGURA 2.5 – Claude Debussy, *Pelléas et Mélisande*, ed. Durand, pag. 149

2.3.1 Note

Per le note si usano i nomi inglesi (sempre minuscolo!), cioè a, b, c, d, e, f, g (la, si, do, re, mi, fa, sol). Il simbolo di ciascuna nota può essere seguito da altri caratteri in un ordine abbastanza libero.

Il primo numero dopo il simbolo della nota indica il valore. I numeri 9, 0, 2, 4, 8, 1, 3, 6 indicano rispettivamente: *breve*, *semibreve*, *minima*, *semiminima*,

croma, *semicroma*, *biscroma*, *semibiscroma*. Non è sempre necessario specificare il valore della nota. All'interno di ciascun *blocco* se il valore non viene specificato **PMX** mantiene il valore della nota precedente.

Il secondo numero rappresenta l'ottava⁶. L'ottava centrale, dal *do* al *si* superiore, si indica con il numero 4, quella inferiore col numero 3 e così di seguito, come si vede in figura 2.6.



FIGURA 2.6 – Indicazione dell'ottava con valori assoluti

L'indicazione di ottava può però essere omessa se la nota si trova nell'ambito dell'intervallo di quarta rispetto a quella che precede⁷. È inoltre possibile, ed anzi consigliabile poiché permette la trasposizione, usare l'indicazione di ottava *relativa*. Se, rispetto alla nota precedente, si resta all'interno della quarta, come s'è detto, non è necessario alcun numero. All'inizio di ciascun *blocco* successivo al primo, se non si specifica il numero d'ottava, **PMX** colloca la nota all'interno della quarta rispetto all'ultima nota del blocco precedente della stessa voce⁸. Per intervalli superiori alla quarta verso l'acuto si usa il segno +, verso il grave il segno -. Se il salto supera l'ottava il segno va raddoppiato⁹. L'esempio di figura 2.7 chiarirà meglio la questione¹⁰.



FIGURA 2.7 – Indicazione relativa dell'ottava

Per la definizione della nota possono essere necessari altri caratteri, che devono essere introdotti senza spazi in mezzo. Eccoli schematicamente (la figura 2.8 ne mostra l'effetto):

⁶Se presenti, le due cifre devono essere scritte in quest'ordine: prima il valore, poi l'ottava.

⁷Se si omette il numero di ottava della prima nota del primo *blocco* **PMX** pone la nota nell'ambito di una quarta dal *do* centrale.

⁸Nel caso si inizi una seconda voce in un pentagramma che nel blocco precedente ne aveva solo una è preferibile indicare il numero d'ottava.

⁹Dalla versione 2.1 è possibile aggiungere un salto oltre la quarta al numero dell'ottava, per esempio c43+. In questo caso, trattandosi di una funzione incompatibile con le versioni precedenti, che non ammettevano tale possibilità, **PMX** stampa un avvertimento.

¹⁰Usando l'indicazione in valori assoluti delle ottave la stessa musica va scritta così: c84 d84 e84 c85 b84 a84 g84 g83 | c26 g23 | c04.

a14 cu cd c8 cdd4 c1 c2 c2 cr2
 c8 c c ca cs cf cn css cff c csc cfc cnc csi cfi cni

FIGURA 2.8 – Caratteristiche delle note

- u** forza verso l'alto (up) la direzione dell'asta
- l** forza verso il basso (lower) la direzione dell'asta
- d** aggiunge il punto (dot) di valore
- dd** aggiunge due punti di valore
- e** sposta la nota a sinistra (left) di una testa di nota
- r** sposta la nota a destra (right) di una testa di nota
- a** non permette che l'asta della nota sia raggruppata mediante i tagli con altre.
- s** pone il diesis (sharp) davanti alla nota
- f** pone il bemolle (flat) davanti alla nota
- n** pone il bequadro (natural) davanti alla nota
- ss ff** raddoppiano le alterazioni
- sc fc nc** pongono le alterazioni tra parentesi
- si fi ni** alterano la nota solo nel file MIDI, senza stampare alcuna alterazione

La posizione dei punti di valore può essere regolata mediante i segni + e -, che devono essere posti dopo d. Il primo valore indica lo spostamento *verticale* rispetto alla posizione normale, mentre il secondo, se presente, indica lo spostamento orizzontale (*vedi* figura 2.9).

c4d+0+1 c8 c4d+2 c8 c4d-2+1 c8 c4dd+2 c1

FIGURA 2.9 – Posizione dei punti di valore

Nello stesso modo è possibile variare la posizione delle alterazioni, ma in questo caso è necessario introdurre tutti e due i valori (ad esempio +1+0 e non solo +1),

relativi rispettivamente allo spostamento verticale e a quello orizzontale. Nella maggior parte dei casi è necessario spostare solo orizzontalmente la posizione degli accidenti. Per questo **PMX** accetta anche i segni < e > seguiti da un numero decimale. Il primo segno indica uno slittamento verso sinistra, mentre il secondo verso destra (*vedi* figura 2.10). Negli accordi è facile che si creino delle sovrapposizioni di accidenti. Se si tratta delle alterazioni di un accordo in una sola voce **PMX** corregge automaticamente le posizioni degli accidenti (*vedi* paragrafo 2.3.3). Se invece ad essere sovrapposte sono le alterazioni di due voci diverse che si trovano sullo stesso pentagramma è necessario correggere la posizione manualmente con il metodo appena indicato¹¹.



FIGURA 2.10 – Posizione delle alterazioni

Per rendere più agevole l'introduzione dei ritmi puntati (con rapporto di valore 3:1) è possibile, anziché includere **d**, introdurre un punto (.) alla fine dei caratteri che descrivono una nota, seguito, *senza spazi*, dalla nota successiva, *senza indicazione di ritmo*. **PMX** stamperà il punto sulla prima nota e dimezzerà il valore della nota successiva. La musica riprodotta in figura 2.11a è stata ottenuta così: e8.g c.a g.f | e4. Assai più semplice di: e8 g1 cd8 a1 gd8 f1 | e4.

Una cosa analoga è prevista per i ritmi ternari le cui note stanno in rapporto di 2:1. In questo caso anziché il punto si usa la virgola (,). Per esempio la musica riprodotta in figura 2.11b si ottiene scrivendo: e4+,d | c,e | d,b | cd. La forma non abbreviata è invece: e4+ d8 | c4 e8 | d4 b8 | cd4.



FIGURA 2.11 – Codificazione semplificata dei ritmi puntati e in rapporto di 2:1

A volte dopo la prima delle due note si devono introdurre i simboli necessari per stampare un abbellimento, una legatura, o altro. In questi casi è possibile utilizzare lo stesso la codificazione abbreviata appena vista, separando però le due note in maniera che la seconda cominci con il punto o con la virgola. Se per esempio abbiamo una semicroma puntata su cui va posto un mordente seguita da una biscroma non possiamo usare la forma abbreviata c1.d, poiché non vi sarebbe posto per l'abbellimento (ot) il cui simbolo deve seguire quello della nota a cui si riferisce. In questo caso si può staccare la seconda nota dalla prima e inserire il mordente fra le due note: c1 ot .d. Perciò il codice di figura 2.11a può essere scritto anche così: e8 .g c .a g .f | e4; quello di figura 2.11b così: e4+ ,d | c ,e | d ,b | cd.

¹¹Per evitare la collisione di un diesis con un altro in un accordo è consigliabile usare il valore 0.85, mentre per evitare collisioni fra due bemolli si suggerisce 0.3.

Un altro tipo di ritmo eccezionale è costituito dai raggruppamenti irregolari di note, come le duine, le terzine eccetera. **PMX** consente l'introduzione di gruppi irregolari da 2 a 24 note o pause. Si deve innanzitutto introdurre il simbolo per la prima nota o pausa nella maniera consueta, attribuendo alla nota *il valore dell'intero gruppo*¹². Si inserisce poi, come al solito *senza spazi*, il simbolo **x** seguito dal numero di note che compongono il gruppo. Infine, questa volta *separati da spazi*, vanno introdotti i simboli delle restanti note del gruppo, senza indicazione di ritmo. Una terzina di crome in 4/4 che sale dal do centrale va scritta così: **c4x3** d e. La musica di figura 2.12, per esempio, è stata ottenuta con il seguente codice:

```
1 1 4 4 0 6 0 0
1 1 20 0.0

t
./
Ab
It64
w167m
% Batt 1
e4x5 f g a b c4x3 g e c2x14 d e f g a b c d e f g a b /

% Batt 2-3
m6868
cd4x2 c- fd4x4 g a b | cd2 Rb /
```

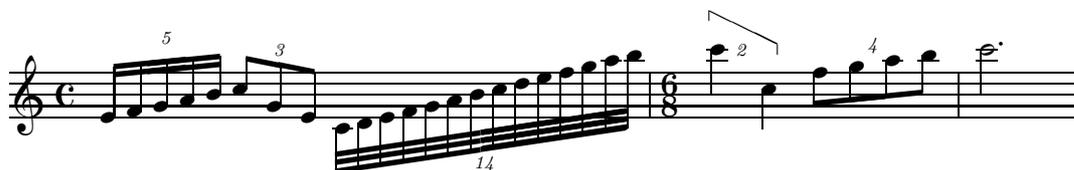


FIGURA 2.12 – Gruppi irregolari di note

Come si vede **PMX** introduce automaticamente il numero delle note del gruppo ponendolo in una posizione conveniente. In questo modo però capita che il numero si trovi talvolta sulle teste e talvolta sulle aste delle note. Se si vuole invertire la posizione del numero si deve far seguire, *senza spazi*, al codice della prima nota il simbolo **n** e poi **f**. Se per esempio non siamo soddisfatti della posizione del numero 4 nella quartina dell'esempio di figura 2.12, possiamo fare in modo che tale numero sia posto sulle aste scrivendo la quartina così: **fd4x4nf** g a b. Se **n** è seguita da un numero intero (*senza segno + o -*) allora **PMX** stamperà quel numero anziché il numero delle note del gruppo. L'opzione **n** serve anche per regolare, orizzontalmente e verticalmente la posizione del numero stampato. Uno o due numeri, sempre preceduti da **+** o **-**, possono seguire **n**. Il primo indica lo

¹²Se non si inserisce alcun valore **PMX** considera valido il valore della nota precedente.

spostamento verticale, in `\internote`, in alto (+) o in basso (-), mentre il secondo, in teste di nota¹³, a destra (+) o a sinistra (-). Se `n` è l'ultimo dei caratteri della nota ed è dunque seguito da uno spazio bianco, allora **PMX** non stampa alcun numero per segnalare il gruppo irregolare. È inoltre possibile modificare lo stile dell'indicazione del gruppo irregolare nei casi in cui non compaiono travature, come capita all'inizio della battuta di 6/8 dell'esempio appena visto. Mediante il comando `AT`, da porsi all'inizio del primo blocco del codice, il numero indicante il gruppo irregolare sarà posto all'interno di una parentesi graffa spezzata (*vedi* paragrafo 2.3.18 a pag. 60). La figura 2.13 mostra come possono essere applicate queste opzioni all'esempio di figura 2.12.

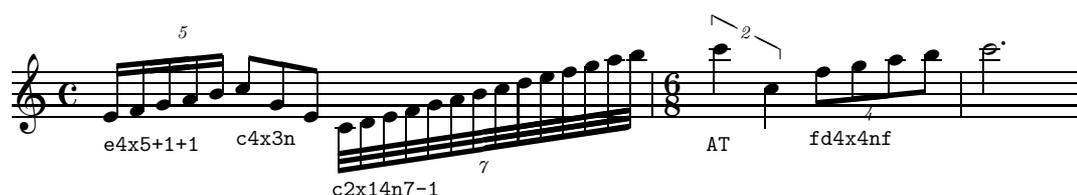


FIGURA 2.13 – Gruppi irregolari di note: controllo del numero da stampare.

In qualche occasione può essere necessario introdurre una nota di valore doppio rispetto alle altre del gruppo irregolare e diminuire di uno il numero totale delle note. Per far ciò basta semplicemente introdurre il carattere `D`. Per aggiungere un punto alla nota raddoppiata anziché `D` si deve aggiungere `F`. Per i ritmi puntati all'interno dei gruppi irregolari è sufficiente aggiungere `d` al codice della nota: **PMX** dimezzerà automaticamente il valore della nota successiva. Un esempio di questi casi particolari è presentato nella figura 2.14.

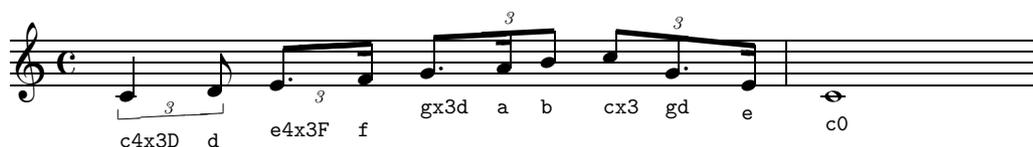


FIGURA 2.14 – Gruppi irregolari di note: casi particolari.

L'unica limitazione nei gruppi irregolari è che *l'ultimo elemento del gruppo non può essere una pausa*.

Un'ultima opzione relativa alla nota è `S` che permette di regolare l'altezza delle aste delle note se non sono legate fra loro. Basta aggiungere, senza spazi, all'interno del codice di una nota, il carattere `S` seguito da un numero (x), senza segno, compreso fra .5 (accorciamento minimo) e 4 (accorciamento massimo). È consentito anche un uso a 'interruttore'. Dopo l'opzione `Sx`: tutte le note saranno accorciate del valore di x sino a che in un codice nota non sarà introdotto il segno `S:`, dopo il quale tutto tornerà normale.

¹³Si tratta di un'unità di misura (inglese: 'notehead') impiegata da **PMX** nelle misure orizzontali.

2.3.2 Pause

Il simbolo **r** indica la pausa. Se si eccettua il fatto che non va ovviamente inserita alcuna indicazione d'ottava, il metodo di introduzione è abbastanza simile a quello delle note. Ad **r** va fatto seguire il numero relativo alla durata, o nulla, se si mantiene il valore della nota o della pausa precedente.

The figure shows two musical staves. The first staff contains the following symbols: **r0**, **r2**, **r4**, **r8**, **r1**, **r3**, **r6**, **rb**. The second staff contains: **r4+3**, **r8-6**, **r**, **r**, **.r**, **rdd**, **r3**, **rp**, **rpo**, **rm15**. A large number **15** is placed above the second staff.

FIGURA 2.15 – Pause

Può seguire **d**, per i valori puntati o il punto (.) o la virgola (,) per i rapporti di valore 3:1 e 3:2, nello stesso modo già visto per le note. Se **r** è seguita da **p**, **PMX** introdurrà una pausa centrata della durata dell'intera battuta; se è seguita da **b** non verrà stampata alcuna pausa, ma il suo valore sarà calcolato dal programma (una funzione assai utile quando vi sono due voci su un solo pentagramma). Se si desidera una pausa della durata dell'intera battuta non centrata si deve aggiungere **o** (es.: **rpo**). **rm** seguito immediatamente da un numero intero serve ad indicare una pausa multipla, cioè valida per più battute. Si tratta di una funzione usata per lo più nelle parti staccate, estratte dalla partitura e, normalmente, la si trova nel codice ricavato dal programma di estrazione delle parti **scor2prt**, tuttavia può essere usata anche in una partitura a più strumenti, avendo però cura di introdurre il numero di battute identico in ogni pentagramma. Per regolare la posizione verticale delle pause va aggiunto **+** o **-** seguito dal valore di **\internote**. Se ci sono due voci sullo stesso pentagramma **PMX** automaticamente sposta le pause della voce superiore in alto di **2 \internote** e quelle della voce inferiore in basso di **4 \internote**. In questo caso, se si desidera porre una pausa nella posizione normale, cioè al centro del pentagramma, è necessario scrivere **r+0**. La figura 2.15 mostra un piccolo campionario di pause.

2.3.3 Accordi

Nella lingua di **PMX** gli accordi sono composti da una nota principale, che va codificata nella maniera già vista, e da una o più note aggiunte per formare l'accordo. Ciascuna nota deve essere separata dalle altre mediante uno spazio bianco. Le note dell'accordo aggiunte alla nota principale devono iniziare con **z**, devono poi seguire il nome della nota, **+** o **-**, se necessario, per l'indicazione dell'ottava¹⁴ e, eventualmente, **e** o **r** per spostare a destra o a sinistra la testa delle note in modo

¹⁴Non è ammessa in questo caso l'indicazione assoluta dell'ottava.

da evitare le sovrapposizioni, anche se ciò non dovrebbe essere necessario, visto che **PMX** sposta automaticamente la testa della nota per evitare collisioni. Non va inserita alcuna durata, né indicato il punto di valore, poiché sono stabiliti dalla nota principale. Solo nel caso in cui si debba modificare la posizione del punto di valore è necessario aggiungere il carattere *d* seguito da uno o due numeri interi preceduti da + o -, come si è già visto per le note (*vedi* pagina 35). Ecco come sono stati ottenuti gli accordi di figura 2.16.

%1-3

```
f22 za zc zf | ff- zb zdf zgf | ef- zg zb zef /
c2 zef zf za | df- zgf zb zdf | b- zdf zef zgn /
```

%4-5

```
dn2- zbn+ zen | cs zes zgs zcs /
bn2- zen zgs | es zgs zcs /
```

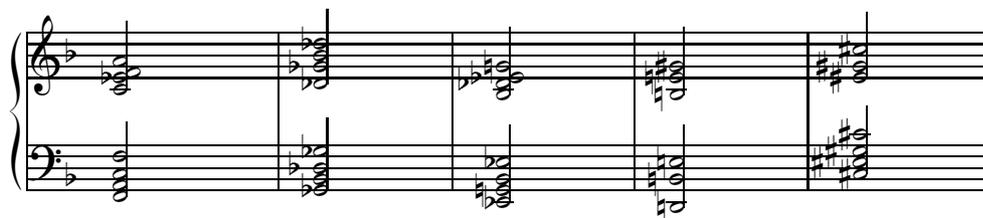


FIGURA 2.16 – Accordi

La direzione dell'asta dell'accordo è regolata dalla nota principale, ma può essere modificata, nella maniera già vista, ponendo *l* o *u* nel codice della nota principale. Anche l'angolatura e la direzione delle aste di più accordi uniti fra loro sono calcolate da **PMX** in base alla posizione della nota principale. Perciò è necessario scegliere come nota principale dell'accordo la più adatta¹⁵. La posizione degli accidenti è stabilita automaticamente dal programma per evitare sovrapposizioni. Se la si vuol modificare manualmente è possibile utilizzare *+i* o *-i*, oppure *<x* o *>x* come si è già visto (*vedi* pagina 36). Se prima dei segni necessari per lo spostamento di un accidente si scrive *A* (per es. *zcsA<.5*), allora tale spostamento sarà *aggiunto* a quello automaticamente calcolato dal programma.

2.3.4 Appoggiature e acciaccature

Le appoggiature e le acciaccature (inglese: *grace notes*), normalmente poste prima della nota a cui si riferiscono, iniziano con il carattere *G*, cui fa seguito una serie di opzioni: un singolo numero intero indica il numero delle note da includere (se non si inserisce alcun numero **PMX** inserisce una sola nota); *m* seguito da un

¹⁵È ovviamente anche possibile modificare l'angolatura delle aste delle note come vedremo fra breve.

numero indica il numero di tagli da porre sulle aste; **s** lega l'abbellimento alla nota principale (non c'è bisogno di altre **s** per indicare la terminazione della legatura); **x** aggiunge il taglio che indica l'acciaccatura e dunque si usa solo quando vi è un'unica nota; **A** (da *after*, cioè 'dopo') o **W** (da *way-after*, 'modo dopo') servono per associare l'abbellimento alla nota che precede, anziché a quella che segue; **X** seguito da un numero decimale (positivo, senza segno) introduce uno spazio aggiuntivo in teste di nota fra l'abbellimento e la nota principale. Ai caratteri opzionali va fatto seguire, senza spazi, l'unico carattere indispensabile (oltre **G** naturalmente), cioè il nome della nota (**a-g**), senza indicazione del valore. Possono essere invece introdotti, nel modo consueto, l'indicazione dell'ottava e gli accidenti. Se le note sono più d'una, quelle successive alla prima vanno introdotte (nome della nota ed, eventualmente, ottava e alterazione) separate da spazi. Le note ornamentali di figura 2.17 sono state ottenute con il seguente codice (omettiamo il preambolo):

```
% Batt 1
G3sm2g++ a b c4 f- G2slAe d c GsxX.5b+ c /
% Batt 2
Ga- g4 Gfs- g c G13sm3d e f g a b c d e f g a b c Rb /
```



FIGURA 2.17 – Appoggiature e acciaccature

2.3.5 Ornamenti

I simboli necessari per gli ornamenti devono essere posti, separati da uno spazio, *dopo* la nota a cui si riferiscono e devono iniziare con **o**. Gli ornamenti previsti da **PMX** sono elencati nella tabella 2.4 e mostrati in figura 2.18.

Il *segno* (**og**) va posto obbligatoriamente solo nella prima voce (la più bassa del sistema). Vi sono due tipi di *staccato* quasi identici, **o.** e **ou**. Nel primo caso (**o.**) il punto viene posto leggermente più in basso rispetto all'altro tipo di *staccato* (**ou**). Tutti gli ornamenti possono essere spostati verticalmente aggiungendo un numero intero positivo o negativo. La *cesura* (**oc**) e il *respiro* (**ob**), che a differenza degli altri segni sono posti `0.5\noteskip` dopo la nota a cui si riferiscono, ammettono anche un secondo numero, positivo o negativo, che ne regola la posizione orizzontale. Il segno **oT** produce un *trillo* con una linea ondulata lunga fino alla nota successiva; **oTx** serve allo stesso scopo, ma regola la lunghezza della linea ondulata a `x\noteskip`; **oT0** stampa *tr* senza linea ondulata; **oTt** stampa solo la linea senza il segno del trillo. I due punti (**:**) servono come comando interruttore: la prima volta che compaiono fanno in modo che l'abbellimento in questione sia aggiunto anche alle note successive dello stesso blocco, la volta successiva indicano

ot om ox o+ ou op o_ o(o) o. o> o^ oes oef oen oes? oe?

oc ob of ofd oT oT oT0 oTt oT1

o. o. o.: o: og

FIGURA 2.18 – Ornamenti

che la nota a cui si riferiscono è l'ultima della serie ad avere quell'ornamento (si veda l'esempio di ripetizione dello *staccato* alla figura 2.18).

mordente	ot
mordente inferiore	om
X	ox
+	o+
staccato	ou
pizzicato	op
tenuto	o_
(prima della nota	o(
) dopo la nota	o)
staccato	o.
accentato	o>
sforzando	o^
accidenti editoriali	oes oef oen
accidente dubbio	oes?
passo dubbio	oe?
cesura	oc
respiro	ob
corona	of
corona inferiore	ofd
trillo	oT, oT α , oT0, oTt
inizio e fine ripetizione	:
segno	og

TABELLA 2.4 – Ornamenti

2.3.6 Legature

PMX accetta come simboli per le legature **s**, **t**, **(**, **)**, **{** e **}**. I primi due vanno posti dopo la nota a cui si riferiscono ed hanno l'effetto di iniziare o di terminare (se già iniziata) una legatura su quella nota. **t** (inglese: tie) produce una *legatura di valore*, cioè pone un arco orizzontale fra due note, mentre **s** (inglese: slur) stampa una *legatura di portamento*, cioè sistema un arco orizzontale o inclinato sopra le teste di due o più note¹⁶. Le parentesi tonde hanno l'identico effetto delle **s**. In questo caso però il segno **(**, che indica l'inizio della legatura, va posto prima della nota a cui si riferisce. Se volessimo dunque legare una coppia di crome ascendenti di grado dal do centrale potremmo indifferentemente usare la forma `c84 s d s` oppure `(c84 d)`. Le parentesi graffe, **{** e **}**, servono per le legature di valore (come **t**) e si usano come quelle tonde. In corrispondenza di una pausa una legatura può terminare, ma *non può mai iniziare*. In tali casi **PMX** predispose un arco orizzontale, ma è possibile variare la posizione verticale sia del suo inizio che della sua terminazione nel modo indicato qui di seguito.

Dopo il segno di legatura è possibile introdurre una serie di caratteri opzionali. Innanzitutto un carattere di identificazione (0-9 oppure A-Z), da porsi sia all'apertura sia alla chiusura della legatura, che è necessario per far sapere a **PMX** quale legatura chiudere se ce ne è più d'una aperta. La direzione della curva può essere forzata verso l'alto con **u** (upper) oppure verso il basso con **l** (lower) o con **d** (down), da porsi nel codice di inizio legatura. **b** serve per ottenere un arco tratteggiato (*vedi* figura 2.20). La posizione sia dell'inizio che della fine della legatura può essere modificata da due numeri preceduti dal segno **+** o **-**. Il primo numero, che deve essere intero, controlla la posizione verticale in `\internote`; il secondo, che può essere decimale, quella orizzontale in teste di nota. Per modificare la forma dell'arco possono essere aggiunti tre ulteriori parametri al simbolo di fine legatura. Il primo, un numero intero che non deve essere zero, altera il punto centrale della curva in `\internote`. Gli altri due, numeri interi fra 1 e 7, preceduti da **:**, modificano il punto di inizio e di fine della curva¹⁷. Due battute (25 e 26) tratte da *Waves*, il primo dei *Poems of the sea* di Ernest Bloch, offrono un buon esempio di legature pianistiche. Sono riprodotte in figura 2.19 e sono state ottenute con il seguente codice **PMX**:

```
2 1 3 4 3 4 0 -5
```

```
1 1 16 0.0
```

```
tt
```

```
./
```

```
Ab
```

```
w170m
```

```
\nobarnumbers\
```

¹⁶Per una discussione più approfondita *vedi* pagina 47.

¹⁷Sono numeri che **PMX** trasforma negli argomenti delle legature di **MuSiXTeX** `\midslur` (un solo numero) o `\curve` (tre numeri).

% Bar 25

```
(A e4x3n g bff cf4x3n b g )A (B e4x3n bf e /
f8+ (C zf+ e- ze+ c4nc- zcnc+ (Dt1 b- D< (Et zb+ /
```

% Bar 26

```
g4x3n an g e4x3n b e g4x3n a g )B+0+0-8 Rb /
b4- )Dt D< zb+ )Et )C+0+0+4:24 dd4- D< (F+1 zd+ o_ e8- D< ze+ )F /
```



FIGURA 2.19 – Legature

Il sistema impiegato da **MusiX_TE_X** per produrre legature è basato su fonti e non sempre produce risultati soddisfacenti, soprattutto quando la distanza verticale fra due note legate supera le 16\internote, come capita nell'esempio di figura 2.20¹⁸.

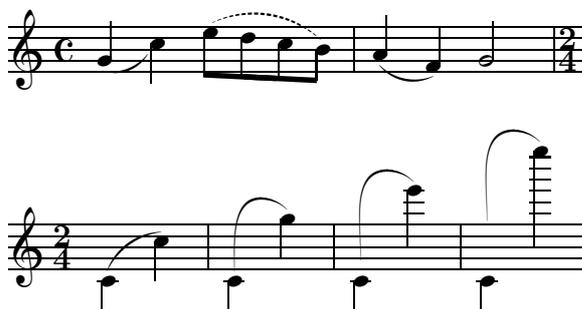


FIGURA 2.20 – Limiti delle legature in **MusiX_TE_X**

Questi limiti possono essere superati facendo uso di legature disegnate in postscript¹⁹. A questo scopo esistono due pacchetti alternativi. Il primo, che chiameremo *Tipo K*, ideato da Stanislav Kneifl, è direttamente supportato da **PMX**. Per attivarlo è sufficiente aggiungere l'opzione **p** al comando **A**, che segue il *preambolo* e precede il *corpo* vero e proprio (*cfr* il paragrafo 2.3.18). L'altro, che indicheremo come *Tipo M*, dal nome del suo autore Hiroaki Morimoto, richiede

¹⁸Tratto dal manuale di **MusiX_TE_X**, p. 39.

¹⁹Ma bisogna considerare che le legature postscript non sono visibili con i normali programmi di visione per file DVI, come **xdvi** o **Yap**. Inoltre, come si vedrà vi sono leggere differenze nel tipo di curva che si produce e ciò rende non immediatamente compatibili le legature postscript e non-postscript.

MetaPost (normalmente presente nelle distribuzioni di $\text{T}_{\text{E}}\text{X}$) e non è direttamente implementato da **PMX**. Tuttavia, essendo stato creato per essere del tutto compatibile con **Mu $\text{S}i\text{X}\text{T}_{\text{E}}\text{X}$** , è facilmente utilizzabile anche con **PMX**, a patto, ovviamente, che il sistema $\text{T}_{\text{E}}\text{X}$ impiegato possenga il pacchetto **MetaPost**²⁰. Tutti e due i pacchetti postscript producono eccellenti legature. Per avere un'idea delle differenze si vedano le figure 2.21, 2.22, 2.23, 2.24, che riproducono le figure 2.19 e 2.20 mediante l'uso dei due pacchetti postscript.



FIGURA 2.21 – Le legature di figura 2.20 con il pacchetto PS *Tipo K*



FIGURA 2.22 – Le legature di figura 2.20 con il pacchetto PS *Tipo M*



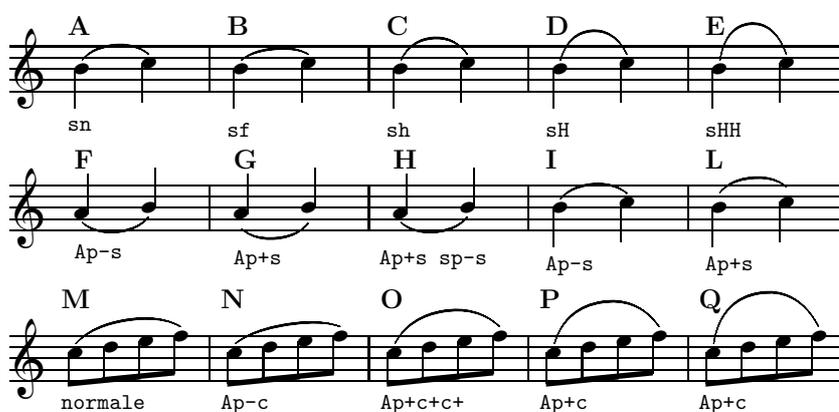
FIGURA 2.23 – Le legature di figura 2.19 con il pacchetto PS *Tipo K*

PMX mette anche a disposizione una serie di opzioni valide *solo* per il *Tipo K*. Il carattere **f**, inserito o nel codice di inizio o in quello di fine legatura, rende più piatto l'arco, mentre, in gradi diversi e crescenti, **h**, **H** e **HH** lo rendono più curvo,

²⁰Per l'uso *vedi* pagina 17.

FIGURA 2.24 – Le legature di figura 2.19 con il pacchetto PS *Tipo M*

come si può vedere nei casi A–E di figura 2.25. `n` può essere usato per ridefinire localmente un arco di curvatura normale quando il suo valore globale sia stato cambiato come si vedrà tra breve.

FIGURA 2.25 – Opzioni di controllo dell'arco nelle legature *Tipo K*

Ci sono alcune opzioni globali, da introdurre mediante il comando `A` (vedi paragrafo 2.3.18, pag. 61), che servono ad attivare o a disattivare i dispositivi anti-collisione delle legature postscript con altri segni musicali. `Ap+s` inserito all'inizio di un blocco attiva tali dispositivi (cfr figura 2.25, casi G ed L), mentre `Ap-s`, sempre all'inizio di un blocco, li disattiva (cfr figura 2.25, casi F ed I).

È possibile ottenere i diversi gradi di curvatura all'interno della scala (da molto piatto a molto curvo) che abbiamo visto poco fa (`f n h H HH`) usando, sempre all'inizio di un blocco, due opzioni di effetto globale: `Ap+c` che aumenta e `Ap-c` che diminuisce la curvatura di un grado. Ad esempio, quando ci troviamo nella posizione normale, se inseriamo il comando `Ap+c` passiamo a quella equivalente ad `h`, se inseriamo `Ap-c` otteniamo il grado corrispondente a `f` e se invece inseriamo `Ap+c+c+c` otteniamo il grado corrispondente a `HH`. Un esempio è mostrato nei casi M–Q di figura 2.25.

È anche possibile variare localmente le impostazioni globali dei comandi `Ap` viste finora, ad esempio, riportando alla normalità con `sn` una legatura dopo il comando `Ap+c` (figura 2.25, caso A), oppure rinunciando agli aggiustamenti anti-collisione, come nel caso H di figura 2.25.

Per le *legature di valore* è possibile sia regolare la posizione della curva apponendo l'opzione `t` dopo `s` o `(`, sia iniziare direttamente la legatura come legatura di valore, cioè col carattere `t` al posto di `s` o `(`. Il risultato è differente se usiamo le legature di **Mu_siX_TE_X** (o quelle postscript *Tipo M*), oppure il *Tipo K*. Nel primo caso `s`, `(` e `t` usati come caratteri iniziali della legatura sono segni perfettamente equivalenti e producono, da soli una legatura di portamento. Per ottenere la posizione della legatura di valore è necessario aggiungere l'opzione `t`, cioè `st`, (`t` o `tt` (cfr figura 2.26). Se si usano invece le legature postscript *Tipo K*, il segno `t` come carattere iniziale produce una legatura di valore (cfr figura 2.27). In questo caso la legatura appare leggermente più piatta, inoltre è posta fra le teste delle note e non su di esse.

FIGURA 2.26 – Legature di valore, **Mu_siX_TE_X** e *Tipo M*FIGURA 2.27 – Legature di valore, *Tipo K*

Per le legature di valore il *Tipo K* mette a disposizione un altro paio di comandi a interruttore, `Ap+t`, `Ap-t`, `Ap+h` e `Ap-h`, da porre come sempre all'inizio di un blocco. La prima coppia di comandi (`Ap+t` e `Ap-t`) ha per le legature di valore lo stesso effetto anti-collisione che `Ap+s` e `Ap-s` hanno per quelle di portamento che abbiamo già incontrato. `Ap+h` si usa quando la legatura si trova a cavallo di un fine riga e fa in modo che la seconda legatura sia disegnata come un semiarco anziché come un arco intero. `Ap-h` disattiva la funzione. La figura 2.28 mostra un esempio di questi comandi.

FIGURA 2.28 – Altre opzioni di controllo nelle legature *Tipo K*

Un altro comando globale, `Ap1`, posto all'inizio di un blocco²¹, attiva una serie

²¹Preferibilmente il primo, visto che si tratta di un comando globale non revocabile.

di aggiustamenti automatici per le legature che si trovano a cavallo del fine riga. In particolare dispone il primo arco (quello in fine riga) in orizzontale, come se si trattasse di una legatura di valore e fa iniziare il secondo arco più a sinistra, vicino alla chiave.

Se ne veda un esempio in figura 2.29.



FIGURA 2.29 – Effetto di `Ap1` nelle legature *Tipo K*

2.3.7 Gruppi ritmici (travature)²²

PMX raggruppa le note automaticamente in base al metro della battuta e stabilisce l'angolo e l'altezza dei tagli. Se si desidera raggruppare in maniera diversa oppure se si intende variare l'angolo o l'altezza dei tagli si devono usare le parentesi quadre []. In questo modo tutte le note all'interno delle parentesi saranno raggruppate. È disponibile anche in questo caso una serie di opzioni. Le aste delle note possono essere forzatamente dirette verso l'alto (upper), verso il basso (lower) o nella direzione opposta a quella stabilita automaticamente da **PMX** (flip). Uno, due o tre numeri opzionali, ciascuno preceduto da un segno, modificano l'aspetto dei tagli. Il primo, che deve essere un numero intero, controlla l'altezza delle aste in `\internote` e può variare da -30 a +30; il secondo, che ammette la stessa gamma di valori (da -30 a +30) modifica l'inclinazione automaticamente stabilita da **PMX**; il terzo è alternativo al primo, ma usa come unità di misura lo spessore di un taglio e consente un valore da 1 a 3 (si usa in genere per aumentare la lunghezza delle aste). Se il risultato prodotto usando quest'ultima opzione non è soddisfacente, è possibile anche usare insieme la prima e la terza opzione (es.: [-1+0+3 cd8 c3 c6 c]). Il segno `m` seguito da un numero compreso fra 1 e 4 impone che il numero dei tagli sia quello espresso dal numero. L'opzione `h` rende

²²Il termine inglese 'beam' ha una diffusione ben maggiore del corrispondente italiano 'travatura', a cui vengono solitamente preferite perifrasi del tipo 'gruppo ritmico' o 'raggruppamento di note'.

orizzontali le aste. Normalmente i gruppi irregolari di note sono legati fra loro e separati dalle altre note. Se si intende unire le aste di un gruppo irregolare insieme a quelle di altre note si devono usare le parentesi quadre []. Le pause, a patto che siano di durata inferiore alla croma, possono trovarsi all'interno delle parentesi quadre []. Il simbolo][, da porsi all'interno di un gruppo di note unite fra i segni [], diminuisce di uno il numero dei tagli in quel punto e subito lo riporta a quello normale.]-[divide in due parti un gruppo di note comprese fra []. Gli esempi di figura 2.30 sono stati ottenuti con il seguente codice:

```
1 1 4 4 0 6 0 -2
1 2 20 0.0

t
./
Ab
It124
w170m
\\nobarnumbers\
% Batt 1 (a)
f1s c f a c fs a c- b g+ b g [l+12-8 b-- g1++ b g ] Rd /
% Batt 2 (b)
[l f1s- c f a ] [u c fs a c- ] [f b g+ b g ] [ b-- g1++ b g ] Rd /
% Batt 3 (c)
[m4 f1s- c f a ] [m1 c fs a c- ] b g+ b g [h b-- g1++ b g ] Rd /
% Batt 4 (d)
[ f1s- c f a ][ c fs a c- ] [ b g+ b g ]-[ b-- g1++ b g ] Rd /
% Batt 5 (e)
[ fs-4x3nf c a+ c1 fs a c- ] b g+ b g b-- g1++ b g Rd /
% Batt 6 (f)
m3434
cd84 c3 c6 c [+0+0+3 cd8 c3 c6 c ] [-1+0+3 cd8 c3 c6 c ] Rd /
```

I raggruppamenti automatici prodotti da **PMX** sono di solito corretti e non richiedono interventi manuali. Può capitare però di dovere intervenire con raggruppamenti forzati quando vi siano metri insoliti o irregolari, come nel seguente caso che riproduce l'incipit dell'*Invenzione a due voci n. 15* di Bach in tempo di 9/16 (figura 2.31). In tali casi si può ricorrere alla funzione di ripetizione del raggruppamento forzato messa a disposizione da **PMX** (:). Includendo i due punti subito dopo il simbolo di apertura di un raggruppamento forzato si otterrà che le gambe delle note successive saranno unite secondo quello stesso modello ritmico fino alla fine del blocco o fino al prossimo segno di raggruppamento forzato, come si vede nell'esempio di figura 2.32, ottenuto con il seguente codice (omettiamo il preambolo).

```
% 1-2
```

FIGURA 2.30 – Gruppi ritmici

[: b1+ f1 f1] g1 f1 f1 b1 f1 f1 | c1+ f1- f1 g1 f1 f1 c1+ f1- f1 /

% 3

[: d3+ b3 d3 f3 b3 f3] d3 f3 d3 b3 d3 b3 f3 b3 f3 d3 f3 d3 Rb /

FIGURA 2.31 – Gruppi ritmici non riconosciuti automaticamente

FIGURA 2.32 – Gruppi ritmici: funzione di ripetizione

Normalmente i raggruppamenti avvengono nello stesso pentagramma, ma talvolta, soprattutto nella musica pianistica (*cfr* l'esempio di figura 2.33), è necessario unire note appartenenti a due pentagrammi successivi.

FIGURA 2.33 – Gruppi ritmici su pentagrammi diversi

Per far ciò è necessario procedere nel seguente modo: 1) si deve iniziare il raggruppamento nel modo consueto (\square) in un pentagramma; 2) si deve terminare il

raggruppamento del primo pentagramma con]j; 3) si deve riaprire il raggruppamento con [j nel pentagramma successivo e 4) lo si deve poi chiudere normalmente con]. È consentito, in questo caso, avere una sola nota in ciascuna delle due parti. Può essere necessario regolare la forma dei tagli uniti nel modo già descritto, inoltre, visto che ciascuna voce deve avere la durata corretta, si dovranno aggiungere delle pause invisibili. Talvolta è necessario mutare la direzione della seconda parte del gruppo mediante u e l. Infine una restrizione: è possibile aprire solo uno di questi raggruppamenti di note fra pentagrammi diversi alla volta. La figura 2.33²³ è stata ottenuta con il seguente codice.²⁴

```
2 1 4 4 0 5 1 0
1 1 20 0.0

bt
./
Ai1.2
Abdv
It124ipipi
w150m
h35m
% Batt 1-2
r4 | [+4-2 c8-- e c+ e c+ e- c a+ ]j st1 | ald2 st ze ze+ r4 Rb /
c4+ zc+ su | bdf2 zbf- a4+ s za- | [ju c8- a+ c e ] c+ r+0 c4 zc- //
rb4 | er0+ | rb0 /
```

2.3.8 Segni di dinamica e di espressione

PMX permette di inserire alcune indicazioni dinamiche con estrema semplicità. Basta aggiungere, dopo la nota su cui si intende porre il segno di dinamica il carattere D immediatamente seguito da uno dei seguenti simboli: pppp, ppp, pp, p, mp, mf, f, ff, fff, ffff, sfz, fp. D< e D> si usano per il crescendo e il diminuendo. Sono comandi a interruttore e vanno posti, come gli altri, dopo la nota a cui si riferiscono. Il primo D< dice a **PMX** che su quella nota va posto l'inizio di una forcella di crescendo, mentre il secondo D< indica il punto in cui la forcella finisce. È necessario che i due segni si trovino all'interno dello stesso blocco. Analogamente, la forcella del decrescendo si ottiene inserendo una coppia di D>. **MuSiX_TE_X** non ammette forcelle più lunghe di 68 millimetri, né consente forcelle inclinate. Per superare queste limitazioni è necessario usare il pacchetto per le legature postscript *Tipo K*. La funzione D può essere anche usata per inserire una stringa di testo nel seguente modo: D"testo". **PMX** automaticamente pone

²³**PMX** non consente di unire le aste delle note appartenenti a due battute diverse, che dunque attraversino la linea di battuta intermedia. In rari e fortunati casi, come quello di figura 2.33 ciò non produce errori. Di norma è invece necessario usare in queste occasioni i comandi **T_EX** in linea.

²⁴**Ai1.2** aumenta la distanza fra i pentagrammi. Per maggiori dettagli *vedi* paragrafo 2.3.18.

il segno di dinamica (anche la forcella del crescendo) sotto la nota e cerca di evitare sovrapposizioni con altri segni. È comunque possibile modificare la posizione del segno mediante uno o due numeri preceduti da segni + o -. Il primo indica lo spostamento verticale in `\internote`, mentre il secondo quello orizzontale in teste di nota. L'altezza delle forcelle del crescendo e del diminuendo si regola con un numero preceduto dal segno, posto nel simbolo di terminazione della forcella. Se si usa il pacchetto postscript *Tipo K*, per consentire l'inclinazione, è possibile introdurre un numero per regolare l'altezza sia all'inizio che alla fine della forcella. Su una nota ci può essere solo una indicazione dinamica predefinita (es.: `Dp` o `Df`), ma possono esserci due comandi per la forcella del crescendo, anche insieme ad una indicazione dinamica predefinita. La figura 2.34 (un frammento della *Patetica* di Čajkovskij) presenta un piccolo campionario di indicazioni dinamiche ed è stata ottenuta con il seguente codice²⁵:

```
1 1 4 4 0 6 0 2
1 1 20 0.0

b
./
Abp
It116icb
w170m
% Batt 1
r2 D"Adagio"+16 r4 e8-- Dpp-1 D< s f D< D>-1 s /
% Batt 2
g4 D>-1 s f2 s D<-1 f8 s D<-1 Dp-1 g s D>-2 /
% Batt 3
a4 s D>-2 D<-2 g2 s D<-2 g8 s Dmp-1 D<-1 a s /
% Batt 4
b4 D<-1 D>-2 asd2 D>-2 D<-2 /
% Batt 5-6
b0 st D<-2 Dsfz-2 D>-2 | b8 st D>-2 Dp-2 r r4 r2 Rb /
```



FIGURA 2.34 – Indicazioni dinamiche

2.3.9 Cambio di chiave

La chiave di ciascuno strumento è definita, come s'è visto, nel preambolo. È però possibile cambiare chiave in qualsiasi punto della partitura con il simbolo `C` seguito

²⁵Viene qui usato il pacchetto postscript *Tipo K*.

da uno dei caratteri che indicano le chiavi visti a pagina 31, cioè t, s, m, a, n, r, b e f, oppure 0, 1, 2, 3, 4, 5, 6, 7. **PMX** introduce la nuova chiave in corpo minore, come si vede nell'esempio di figura 2.35.

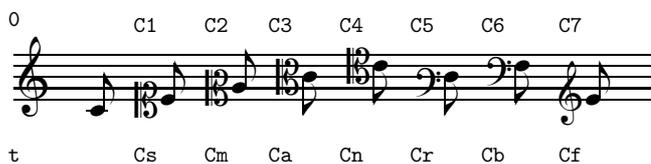


FIGURA 2.35 – Cambio di chiave

2.3.10 Arpeggi

L'arpeggio si ottiene ponendo il simbolo ? dopo la prima e dopo l'ultima nota dell'accordo da arpeggiare. Se sono presenti accidenti la linea ondulata potrebbe sovrapporsi agli accidenti. In tal caso essa può essere spostata facendo seguire al segno di inizio arpeggio un numero decimale negativo (es.: ?-.7). La figura 2.36 mostra alcuni esempi di arpeggio. Ecco il codice (senza preambolo) da cui è ricavata:

```
% 1-2
e2- ? zg zb ds- ?-.7 zf zb | e2- ? zg zb b- ?-.7 zds zf za zb /
e2 zg zb ? f zb ? | e2- zg zb ? ds- zf za zb ? /
```

```
% 3
e0- ? zg zb Rb /
e0- zg zb ? /
```

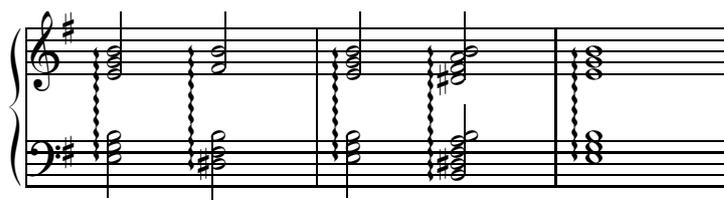


FIGURA 2.36 – Arpeggi

I comandi appartenenti al *corpo* del codice **PMX** visti finora si riferiscono o sono collegati a una nota. Esistono però anche comandi più generali che riguardano tutte le voci, come i segni di ritornello, i comandi di impaginazione e così via. Li vedremo nelle pagine che seguono. A parte casi eccezionali, vanno posti nella prima linea di musica (sono relativi dunque alla voce più in basso nel sistema) e, quando si estraggono le parti, **scor2prt** li aggiunge automaticamente in ogni singola parte.

2.3.11 Ritornelli e aste di battuta

I ritornelli e gli altri casi particolari di aste di battuta si ottengono con il carattere **R** seguito dalle seguenti opzioni:

- l ritornello a sinistra
- r a destra
- lr doppio
- d doppia linea
- D doppia linea (sottile e grossa)
- d1 Doppia linea con ritornello a sinistra
- b linea singola in fine di movimento
- z linea invisibile in fine di movimento

Il comando deve essere posto nella prima voce e non deve essere ripetuto nelle altre voci. La figura 2.37 mostra un esempio dell'uso di questi simboli.

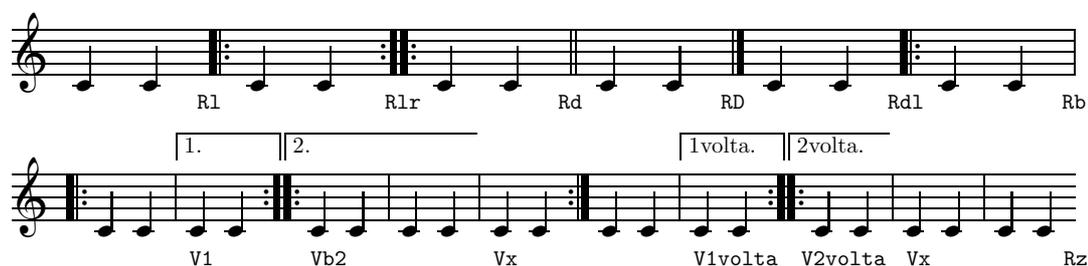


FIGURA 2.37 – Ritornelli e volta

2.3.12 Volta

Il simbolo **V** produce la volta. Va introdotto all'inizio della battuta e solo alla prima voce. Le battute contenenti la volta devono essere poste su blocchi diversi, poiché è consentito introdurre solo una volta per blocco. Il carattere **V** solo, o seguito da qualsiasi testo che non contenga spazi e non inizi con **b** o **x**, produce il segno di inizio della volta. Il caso più comune è **V1** che indica la prima volta e produce una linea chiusa che sovrasta il numero 1. **Vb**, seguito opzionalmente da una stringa di testo, indica la seconda volta (o chiusura della volta). **V2**, ad esempio, produce una linea, aperta in conclusione, che sormonta il numero 2. **Vx** segnala la fine della volta, cioè la prima battuta successiva alla volta. Se ne veda un esempio in figura 2.37.

2.3.13 Cambio di metro

Ogni cambio di metro deve essere effettuato all'inizio di un blocco. Per ottenerlo si devono introdurre quattro numeri preceduti da **m**, secondo l'uso già visto nel preambolo (*cfr* paragrafo 2.2, pagina 30). Vi sono due sistemi alternativi di codifica.

1. Inserire il carattere **m** seguito da quattro numeri, che rappresentano il *numeratore logico*, il *denominatore logico*, il *numeratore stampato* e il *denominatore stampato*. Per esempio, per il 3/4 si scriverà **m3434**, per il tempo a cappella **m4406** e così via. Se si introduce **1 PMX** lo intende come la prima cifra di un numero di due cifre, quindi si deve usare **1** solo in casi come **m128128** (12/8) e non per metri come 1/8 o 1/4. In questi casi è necessario usare **o** al posto di **1**, dunque **mo8o8** o **mo4o4**. Si ricordi che i valori utilizzabili come denominatori logici sono quelli corrispondenti ai valori delle note e che dunque l'intero non è **1**, ma **0**, perciò il metro 2/1 sarà indicato **m202o**.
2. Inserire il carattere **m** seguito da quattro numeri secondo le regole già viste, ma separati da una sbarra (/). In questo modo non è necessario usare **o** al posto di **1**. Dunque 1/8 è **m1/8/1/8**, 12/8 è **m12/8/12/8** e 2/1 è **m2/0/2/1**.

2.3.14 Cambio di tonalità

In ogni momento è possibile cambiare la tonalità e, conseguentemente, l'armatura di chiave. Basta inserire **K+0**²⁶ seguito da un altro numero intero che stabilisce quante alterazioni dovrà avere la nuova tonalità. Come si è già osservato a proposito del *preambolo*, se il numero è positivo, la tonalità conterrà dei diesis, se è negativo, dei bemolli. La figura 2.38 mostra alcuni esempi di cambio di tonalità²⁷.

FIGURA 2.38 – Cambio di tonalità

2.3.15 Trasposizione

PMX permette di trasporre agevolmente e semplicemente un intero brano, ma non consente di trasporre solo alcune voci di una partitura. È però possibile trasporre solo alcune parti quando le si estrae con **scor2prt**. Inoltre con un l'aiuto del pre-processor **ABCPP** si può superare abbastanza facilmente tale limitazione. Dedicheremo una sezione a parte al problema della trasposizione (*vedi* paragrafo 4.3.3). Per ora concentriamoci sulle funzioni presenti in **PMX**.

Per trasporre un brano è necessario usare il sistema di codifica *relativo* delle alterazioni, un sistema peraltro spesso impiegato nella notazione tradizionale da

²⁶Nel cambio di tonalità il primo numero deve essere sempre +0.

²⁷Ovviamente, il cambio di tonalità non è una trasposizione e le note devono essere scritte esattamente come sono.

molti musicisti. In questo sistema il diesis innalza e il bemolle abbassa di un semitono la nota in oggetto e tale alterazione si somma a quella presente in chiave. Per esempio, in tonalità di re maggiore (due diesis in chiave) per ottenere un fa naturale non si dovrà scrivere $fa\sharp$, ma $fa\flat$ e in tonalità di re minore (un bemolle in chiave) per ottenere il si naturale non si dovrà scrivere $si\flat$, ma $si\sharp$. Il bequadro invece stampa l'alterazione naturale della nota davanti alla quale è posto; per esempio, in tonalità di sol maggiore (un diesis in chiave) un bequadro davanti a sol produrrà $sol\flat$ e davanti a fa $fa\sharp$. Per avvertire **PMX** che le alterazioni sono state codificate in modo relativo è necessario usare il comando **Ar** subito dopo il preambolo²⁸. All'inizio del primo blocco va poi posto **K** seguito da due numeri interi, positivi o negativi. Il primo indica la distanza di trasposizione in `\internote`, in alto (segno +) o in basso (segno -). Il secondo numero indica la nuova armatura di chiave (*cfr* paragrafo 2.3.14). Come esempio prendiamo una scala di re minore, ottenuta dal codice che segue e riprodotta in figura 2.39 e proviamo a trasporla prima un tono, poi un semitono sopra.

```
1 1 4 4 0 6 0 0
11 20 0.0
```

```
t
./
Abr
It64
w160m
\\nobarnumbers\
d8 e f g a bs cs d | d cn bn a g f e d Rb /
```



FIGURA 2.39 – Scala di re minore non trasposta

Per ottenere la trasposizione di un tono basta aggiungere il comando di trasposizione, **K+1+1**, all'inizio del primo (e nel nostro caso unico) blocco nel seguente modo (trascriviamo solo il blocco, visto che il resto è identico all'esempio precedente):

```
K+1+1 d8 e f g a bs cs d | d cn bn a g f e d Rb /
```

La prima cifra avverte **PMX** che il brano va scritto una `\internote` sopra (re=mi, mi=fa ecc.) e la seconda indica l'armatura di chiave della nuova tonalità di *mi minore* (+1= un diesis). In figura 2.40 se ne vede l'effetto.

Per trasporre la stessa scala di un semitono in alto la prima cifra dovrà essere -0 (così **PMX** capisce che le note hanno lo stesso nome) e la seconda dovrà indicare

²⁸r è una delle opzioni di **A**, comando che si vedrà in dettaglio più avanti (*cfr* paragrafo 2.3.18), ma che abbiamo già incontrato.



FIGURA 2.40 – Scala di re minore trasposta di un tono

il numero di diesis necessari per la tonalità di *re# minore* risultante, cioè +6. Ecco il codice del blocco:

```
K-0+6 d8 e f g a bs cs d | d cn bn a g f e d Rb /
```

La figura 2.41 ne mostra il risultato.



FIGURA 2.41 – Scala di re minore trasposta di un semitono

2.3.16 Titoli e testi sopra e sotto il pentagramma

PMX mette a disposizione tre elementi per il titolo:

- Tt: titolo del brano (centrato)
- Tc: nome del compositore (giustificato a destra)
- Ti: nome dello strumento (giustificato a sinistra)

Ciascuno di questi comandi deve apparire solo all’inizio del primo blocco, in una riga a sé. La riga che segue il comando è riservata al testo che sarà stampato. Sebbene tale testo debba essere contenuto in una sola linea, è consentito nella stampa l’*‘a capo’*, mediante l’introduzione del comando `\\` nel punto in cui si vuole concludere una linea e iniziare la seguente. Normalmente Ti si usa nelle parti staccate ed è prodotto automaticamente da `scor2prt`. Se si desidera lasciare dello spazio bianco fra il blocco del titolo e il primo sistema musicale è possibile farlo aggiungendo a Tt (senza spazi) un numero di una o due cifre, che rappresenta lo spazio in `\internote` da inserire fra il titolo e il primo sistema.

In maniera analoga funzionano i comandi h e l, i quali, posti all’inizio di un blocco (non necessariamente il primo), accettano un testo scritto nella riga successiva e indicano a PMX di stamparlo sopra (h) o sotto (l) il pentagramma più in alto del sistema, nella prima battuta del blocco in cui il comando è stato introdotto. h può essere seguito da un numero, positivo o negativo, che indica lo spostamento in `\internote` del testo rispetto alla posizione predefinita. Si tratta di una funzione utile per i movimenti, i sottotitoli e altre indicazioni di carattere testuale. Ecco un esempio di codice contenente questi comandi:

L'opzione **c**, che deve essere introdotta per ultima, permette di inserire un breve testo che sarà stampato centrato su tutte le pagine *dopo la prima*. Se dopo **c** appare uno spazio, allora **PMX** stamperà su ogni pagina il nome dello strumento così come appare nel comando (che deve perciò essere presente) **Ti**. Se dopo **c** appare qualsiasi stringa di caratteri eccetto " , senza spazi, allora sarà tale stringa ad essere stampata. Se infine è necessario che il testo da stampare contenga degli spazi (sia cioè composto da più parole), basta racchiudere il testo fra virgolette ("...").

Il programma di estrazione delle parti **scor2prt** ignora il comando **P**, ma se si desidera introdurlo, lo si può fare mediante il comando **%!P2** (inizia la numerazione dal numero 2 su tutte le parti) o **%1P2** (inizia la numerazione dal numero 2 solo nella parte 1)²⁹.

2.3.18 Comandi di carattere generale attraverso A

Il simbolo **A** permette di inserire una serie di comandi che modificano aspetti generali della partitura. È possibile ad esempio scegliere la grandezza delle alterazioni, decidere se usare o meno le legature postscript tipo **K**, regolare la distanza fra i sistemi ed altre cose simili. Le diverse opzioni devono seguire **A** o l'opzione precedente senza spazi. I comandi possono essere dati separatamente (**Ab**, **Ap**) o insieme (**Abp**) e, salvo diversa indicazione, devono essere posti all'inizio del primo blocco.

Alterazioni

Ab, **As**: Stampa le alterazioni tutte grandi (**b**) o tutte piccole (**s**). Impostazione predefinita: grandi o piccole a seconda dello spazio disponibile.

Ar: impiega la notazione relativa degli accidenti. Da usarsi per la trasposizione (*vedi* paragrafo 2.3.15).

Posizione dei punti di valore nei pentagrammi con due voci

Ad: se in un pentagramma vi sono due voci, inserendo questa opzione i punti di valore della voce inferiore verranno stampati o in centro (se la nota è nello spazio) o in basso (se è sulla linea).

Posizione delle pause nei pentagrammi con due voci

AK: posto all'inizio di un blocco, attiva (la prima volta che compare o se la funzione non è attiva) o disattiva (la volta successiva o se la funzione è già attiva) il posizionamento automatico delle pause in base al contesto.

AKL: come il precedente, ma fa in modo che il calcolo della posizione sia effettuato in base alla nota precedente ('Look Left', cioè 'guarda a sinistra') anziché in base a quella successiva.

²⁹ Cfr paragrafo 2.5.

Spazio dopo la linea di inizio battuta

Aa: stabilisce lo spazio dopo la linea di inizio battuta. Il comando deve essere seguito da un numero decimale, indicante un valore di `\afterruleskip` (spazio dopo la linea di battuta) diverso da quello predefinito, che è uguale a `1\elemskip`³⁰.

Giustificazione verticale della pagina

AI, Ai: fattore di `\interstaff`.³¹ Può essere inserito all'inizio di qualsiasi blocco. Modifica lo spazio verticale fra i pentagrammi di un sistema per tutta la partitura (**I**) o per la pagina corrente (**i**).

Ae: uniforma la distanza fra i sistemi. Normalmente **PMX** distanzia i sistemi nella pagina in base allo spazio disponibile tenendo conto del contenuto musicale. Con questa opzione i sistemi saranno posti alla stessa distanza l'uno dall'altro.

Av: posto all'inizio di un blocco, disattiva il comando che prevede che quando lo spazio bianco fra i pentagrammi supera i `20\interstaff` tali pentagrammi siano compattati in alto nella pagina. Un identico comando riattiva la funzione.

Pentagrammi in corpo minore

AS: avverte **PMX** che alcuni pentagrammi sono in corpo minore. **PMX** non permette di definire pentagrammi di diversa grandezza, cosa che però è possibile mediante l'inserimento di comandi `TEX` in linea (vedi paragrafo 4.1.2, pag. 114). Questo comando consente a **PMX** di calcolare esattamente lo spazio occupato dalle alterazioni nei pentagrammi in corpo minore. Dopo **AS**, senza spazi, devono comparire tanti caratteri quante sono le voci; ciascun carattere deve essere `-` oppure `0` per indicare rispettivamente un pentagramma in corpo minore o di grandezza normale.

Nome delle parti estratte

AN: permette di inserire un nome arbitrario per uno o più strumenti nelle parti estratte da `scor2prt`. Normalmente il nome della parte è preso dal preambolo di **PMX**; con questo comando è invece possibile stabilire un nome diverso solo per le parti. **AN** deve essere seguito dal numero della parte e dal nome racchiuso tra virgolette (es.: `AN2"Viola"`).

Stile dell'indicazione dei gruppi irregolari senza travature

AT: permette di variare lo stile dell'indicazione dei gruppi irregolari in cui non vi sono travature. Inserendo questo comando il numero indicante il gruppo irregolare non sarà posto sotto la parentesi quadra che sovrasta le note, ma all'interno di essa, che però apparirà interrotta (vedi l'esempio di figura 2.13 a pag. 38).

³⁰`\elemskip` è l'unità di misura orizzontale basilare in **Mu_siX_TE_X**.

³¹ Vedi in proposito anche il paragrafo 4.1.4.

Inclusione di file

AR: permette di includere file esterni contenenti comandi **PMX** corretti. Vi sono due tipi di inclusione: *globale* e *normale*. Per il tipo *globale* si rimanda al manuale di Don Simons [16, pag. 21]. L'inclusione *normale* si ottiene semplicemente ponendo, preferibilmente all'inizio di un blocco, il comando **AR***nomefile*. Il nome del file da includere (*nomefile*) può avere qualsiasi estensione e deve seguire senza spazi il comando **AR** (es.: **AR**mov2.txt). Il file deve essere un normale file di testo contenente comandi **PMX** corretti. Si può includere un numero illimitato di file, si può includere più volte lo stesso file, ma nel file incluso *non può* essere contenuta un'altra inclusione di file via **AR**.

Comandi per le legature postscript tipo K

Ap: attiva le legature postscript tipo K. Senza altre apzioni le legature presenti nella partitura saranno disegnate in postscript anziché mediante fonti (*vedi* pag. 44 e seguenti). Le opzioni che seguono consentono di definire alcuni aspetti che riguarderanno tutte le legature della partitura o almeno quelle di un intero blocco. Per le regolazioni 'locali' di ciascuna legatura valgono le indicazioni già fornite alle pagg. 45 e seguenti.

Ap+s, **Ap-s**, **Ap+t**, **Ap-t**: posti all'inizio di un blocco, attivano (+) o disattivano (-) il dispositivo anti-collisione per le legature di portamento (**s**) e di valore (**t**), come si è già visto alle pag. 46 e seguenti (*vedi* anche gli esempi delle figure 2.25 e 2.28).

Ap+h, **Ap-h**: posti all'inizio di un blocco, attivano (+) o disattivano (-) le semilegature per la seconda parte delle legature di valore (*vedi* pag. 47 e la figura 2.28).

Ap1: aggiusta automaticamente la posizione degli archi che si trovano a cavallo del fine riga, come si vede nella figura 2.29 a pag. 48.

Ap+c, **Ap-c**: posti all'inizio di un blocco, aumentano (+) o diminuiscono (-) la curvatura degli archi delle legature secondo la sequenza **f n h H HH** da piatto a molto curvo. È anche possibile saltare da un grado all'altro della scala ad esempio con **Ap+c+c** o **Ap-c-c-c** (*vedi* pag. 46 e la figura 2.25).

Aph: copia l'intestazione presente in **psslur.pro** all'inizio di ogni pagina anziché solo all'inizio della prima pagina. Ciò è necessario quando si estraggono pagine della partitura con **dviselec** o simili programmi.

2.3.19 Spazio addizionale orizzontale nel testo musicale

PMX e **MusiX_{TEX}** calcolano lo spazio e giustificano le righe musicali. Normalmente il risultato è eccellente, ma talvolta, per evitare piccole sovrapposizioni di simboli o per ottenere un risultato particolare, è possibile che ci sia bisogno di alcune correzioni nella giustificazione orizzontale. A questo scopo si deve usare il comando **X**, grazie al quale è possibile spostare un solo segno musicale, un gruppo

di segni o tutti gli elementi presenti in un sistema. Se è presente **S** lo spostamento riguarda solo la nota o la pausa successiva. I due punti (:) segnalano l'inizio di un gruppo o la fine (in questo caso i due punti non saranno accompagnati da altre opzioni, quindi: **X:**). Se non sono presenti né **S**, né **:**, allora **PMX** aggiungerà uno spazio bianco in tutto il sistema. Lo spostamento è determinato da un numero decimale —che deve essere inserito dopo **X** e che può essere preceduto dal segno - (spostamento indietro)—, che indica lo spostamento in teste di nota. Se il numero è seguito da **p**, allora lo spostamento è in punti. **scor2prt** non copia nelle parti staccate il comando **X**, ma se si desidera che ciò avvenga, allora si deve aggiungere l'opzione **P**, se si desidera che la funzione sia usata *solo* nelle parti staccate e non nella partitura, oppure **B**, se si vuole che la funzione sia usata tanto nelle parti quanto nella partitura.

2.3.20 Spazio minimo fra le note

Lo spazio minimo consentito da **PMX** fra due note è di 0.3 teste di nota. Questo valore può essere modificato mediante il comando **W** seguito da un numero decimale compreso fra **.1** e **.9** (nuovo valore in teste di nota).

2.3.21 Dimensioni della pagina

Le dimensioni predefinite della pagina sono 740x524 punti (26.1x18.48 centimetri, B5JIS). Per modificare le dimensioni della pagina basta inserire, all'inizio del primo blocco il comando **hnu** per l'altezza della pagina e **wnu** per la larghezza, dove **n** è un numero decimale che indica la nuova dimensione e **u** l'unità di misura, che può essere **i** (inches), **m** (millimetri) o **p** (punti).

2.3.22 Comandi di impaginazione

Il nono e il decimo dei dodici numeri del preambolo stabiliscono il numero delle pagine e il numero totale dei sistemi nella partitura. **PMX** calcola in base al contenuto musicale quante battute possono stare in ciascuna riga e ottiene così l'impaginazione complessiva. È però possibile andare a capo in ogni punto in cui lo si desideri mediante il comando **Ln**, dove **n** è un numero intero minore o uguale al numero totale dei sistemi indicato nel preambolo. Il comando va introdotto all'inizio di un blocco e indica che in quel punto deve iniziare il sistema indicato dal numero **n**. Per esempio **L5**, posto all'inizio di un blocco in corrispondenza con la battuta 30, ordina a **PMX** di porre quella battuta all'inizio del quinto sistema della partitura.

Con il comando **LnPm** si ottiene un'interruzione di pagina. Va anch'esso posto all'inizio di un blocco e indica che la prima battuta di quel blocco si troverà all'inizio del sistema **n** e all'inizio della pagina **m**. **m** deve essere un numero intero minore o uguale al numero totale di pagine stabilito nel preambolo e deve essere sempre preceduto da **L**.

In un solo file **PMX** è possibile porre più di un movimento. Per segnalare il punto in cui inizia il nuovo movimento si usa il comando `LnM` o `LnPmM`³² (se il nuovo movimento inizia su una nuova pagina). `M` può essere seguito dalle seguenti opzioni:

- `+i` introduce uno spazio verticale aggiuntivo di `i \internote`
- `ix` stabilisce un nuovo valore di rientro del primo sistema, indicato da `x` che è un numero decimale rappresentante una frazione dell'intera linea
- `c` ordina a **PMX** di continuare la numerazione delle battute da dove si era conclusa anziché ricominciare da 1
- `r+,-` ristampa (+) o no (-) i nomi degli strumenti davanti al primo sistema (valore predefinito: li ristampa solo se il numero degli strumenti è cambiato)
- `ni` modifica il numero di strumenti (`i` = nuovo numero di strumenti. N.B.: se il numero è di due cifre deve essere *preceduto* da due punti). L'opzione deve essere seguita, senza spazi, da una lista di numeri, ciascuno indicante il numero di strumento presente nel nuovo ordine e da un'altra lista, sempre senza spazi, indicante, sempre nello stesso ordine, le chiavi di ciascuno strumento.

Un esempio. In una partitura di quattro strumenti (violoncello, viola, violino 2, violino 1) la viola tace per tutto il secondo movimento, che può dunque essere stampato con un pentagramma in meno. Per ottenere ciò va inserito all'inizio del secondo movimento il comando `L25Mn3134btt`, che significa: alla linea 25 (`L25`) inizia un nuovo movimento (`M`) contenente tre strumenti (`n3`), che della lista originale sono i nn. 1, 3 e 4 (`134`), le cui chiavi sono, nell'ordine, basso, violino e violino (`btt`). Se nel movimento successivo, poniamo alla linea 43, la viola riprende a suonare, allora inseriremo all'inizio del blocco il comando `L43Mn41234batt`.

Va precisato che il numero di strumenti può essere diminuito rispetto a quello del movimento iniziale, ma *non aumentato*. Per aumentarlo è necessario introdurre una pagina vuota con il numero massimo di strumenti e usare l'opzione `M` prima per ridurre tale numero nel primo movimento, poi per aumentarlo nel movimento successivo. Il procedimento, come si vede, è un po' macchinoso, perciò è preferibile in questi casi creare un codice **PMX** per ciascun movimento e poi impaginare i diversi movimenti con `TEX`³³.

Si possono vedere esempi di questi comandi e dei loro effetti nell'impaginazione nei paragrafi 5.1 e 5.4.

³²Se è presente, `P` deve sempre precedere `M`.

³³*Cfr* paragrafo 4.1.1.

2.3.23 Misure incomplete

Se, come capita ad esempio in corrispondenza dei ritornelli o della fine di una sezione di un brano che comincia con un levare, è necessario introdurre una misura incompleta, è possibile farlo effettuando un cambio di metro invisibile, secondo i criteri già visti nel paragrafo 2.3.13. Per esempio se vogliamo dividere una battuta di 4/4 in modo che una parte di 3/4 si trovi alla fine di una riga e la parte restante di 1/4 all'inizio della riga successiva, come levare della battuta che segue, dobbiamo introdurre tre cambiamenti di metro invisibili. Il primo, per la battuta di 3/4, sarà m3400, il secondo, per il levare di 1/4 sarà mo400 e l'ultimo, per tornare alla normalità, sarà m4400.

2.3.24 Basso continuo

PMX consente di inserire in maniera assai semplice la cifratura del basso, come appare dal seguente esempio di codice, che riproduce una parte del basso continuo della celebre aria monteverdiana, *Pur ti miro*, che conclude l'*incoronazione di Poppea*. L'effetto è visibile nella figura 2.43.

```
1 1 6 4 2 1 0 1
0 6 20 0.0
```

```
b
```

```
./
```

```
Ab
```

```
It64
```

```
w165m
```

```
% batt. 1-4
```

```
gd2 fd 6 | ed 5 x126 dd 4 x123 | gd fd 6 | ed 6 dd 4 x123 /
```

```
% batt. 5-7
```

```
gd2 fd 6 | ed 64 x123 dd | gd fd 6 /
```

```
% batt. 8-10
```

```
ed2 64 x123 dd | gd 03 fd | ed dd 4 x12# /
```

```
% batt. 11-15
```

```
gd2 02 fd | ed 6 dd 7 | gd fd 6 | gd ad 4 x12# | dd- gd /
```

```
% batt. 16-17
```

```
cd- dd 64 x145 x12_00.2 x12_# | gd fd 01 Rd /
```

Per introdurre la cifratura del basso basta aggiungere i numeri necessari, dall'alto al basso, *dopo* la nota alla quale si riferiscono, per esempio 64 (6, sopra il 4) o 642 (6, sopra il 4, sopra il 2). Le alterazioni vanno poste prima del numero a cui si riferiscono. Per il diesis si usa #, per il bemolle -, per il bequadro n.

FIGURA 2.43 – Cifratura del basso

Per esempio 6, sopra 4 con il diesis, sopra 2 si scrive $6\#4_2$; 6 con il bemolle -6 e solo il bemolle indicante la terza $-$. **PMX** sistema i numeri subito sotto le note e calcola la posizione in modo da evitare sovrapposizioni. Talvolta dopo un accordo in quarta e sesta ricorre un accordo allo stato fondamentale con la terza maggiore. In questo, come in altri simili casi, dopo il 6_4 sulla prima nota è necessario allineare il diesis indicante la terza maggiore dell'accordo seguente sulla riga più bassa, cioè di fianco al 4. Per far ciò è necessario introdurre due cifre, una prima cifra invisibile, rappresentata dal segno $_$ e poi il diesis, cioè: $_ \#$ (cfr la battuta 16 di figura 2.43). In pratica quando una cifra è preceduta dalla lineetta in basso essa viene posta in una posizione verticale inferiore di 4 \internote rispetto a quella normale. La posizione delle cifre prestabilita dal programma può essere anche innalzata aggiungendo alla cifra $+$ e un numero intero (in \internote). Combinando i due ultimi comandi è possibile avere un controllo completo della posizione dei numeri.

Se una cifra si trova non in corrispondenza di una nota, ma dopo, allora si devono far precedere a tale cifra tre caratteri: x seguito da due numeri interi. Il primo numero è un moltiplicatore, il secondo indica la durata in base ai valori consueti delle note (2, 4, 8, 1, 3). Se ad esempio si vuole introdurre un 6 a distanza di una semiminima (1/4) dopo la minima che precede si deve scrivere $x14_6$ (vedi anche le batt. 2, 4, 6, 8, 10 e 14 di figura 2.43). Per introdurre la linea continua, che indica che una nota viene tenuta, si usa 0 seguito da un numero. Il numero rappresenta la lunghezza della linea in \noteskip. La linea inizia alla sinistra della nota a cui si riferisce, in corrispondenza della nota precedente (cfr le batt. 9, 11, 16 e 17 di figura 2.43). Se sotto la linea continua deve essere posta un'altra cifra, la si deve separare mediante $:\cdot$. Ad esempio una linea sopra un 3 va codificata così: $01:3$.

È possibile introdurre la cifratura in due parti: nel pentagramma 1 (il più basso nell'ordine) e in un altro pentagramma qualsiasi.

Dalla versione 2.520 **PMX** permette l'introduzione di alcuni caratteri speciali (solo per grandezze di 16 o 20), indicanti le forme alterate dei numeri 2, 4, 5, 6 e

9. Deve però essere installato nel sistema il pacchetto **figbass** contenente i font (lo si trova sul **Werner Icking Music Archive**, <http://icking-music-archive.org>),). Per selezionare tali caratteri è sufficiente aggiungere **s** alla cifra, come si vede nell'esempio riprodotto in figura 2.44, ottenuto con il seguente codice (omettiamo il preambolo):

```
g43 4s g4 6s g4 9s g4 5s | g4 2s g4 6s4s g4 4s2s g4 6s5s Rb /
```

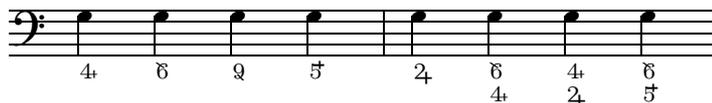


FIGURA 2.44 – Cifratura del basso: caratteri speciali

Se, per qualche motivo, si desidera che la cifratura del basso presente in un codice `.pmx` venga ignorata, basta aggiungere **F** all'inizio del *corpo* del codice. Questa opzione, nella forma condizionale `%1F` (vedi il paragrafo 2.5) è assai utile quando si estrae la parte del basso da destinare, per esempio, al violoncello, in cui non sono necessari i numeri.

Se il brano è trasposto i numeri non vengono modificati in alcun modo. In questo caso saranno dunque necessari aggiustamenti manuali.

PMX non prevede la possibilità di introdurre numeri di due cifre per il basso. Se, per restar fedeli alla cifratura che talvolta si trova in alcune edizioni antiche, si vogliono introdurre numeri con due cifre, lo si deve fare o utilizzando il comando `D" . . . "`³⁴, oppure introducendo, mediante comandi `TeX` in linea, i numeri servendosi della macro predisposta da **PMX**. Nel file `pmx.tex` per l'inserimento dei numeri è stata predisposta la definizione `\Figu{m}{n}` dove `m` è un numero che regola la posizione verticale ed `n` è la cifra. Le alterazioni vanno indicate con `\sharpfig`, `\flatfig` e `\natfig`. Per inserire più facilmente cifre superiori al 9 possiamo predisporre una definizione che accetti due numeri (i valori per le due variabili `m` e `n`) e centri le due cifre sotto la nota. Eccola.

```
\def\BN#1#2{\loffset{.3}{\Figu{#1}{#2}}}
```

Per introdurre i numeri nella parte del basso si userà il comando `\BN{m}{n}`. Se ad esempio vogliamo un 11 si scriverà `\BN{0}{11}`; se invece vogliamo un 10 preceduto da un diesis si dovranno usare due comandi in successione: `\BN{-1}{\sharpfig}\BN{0}{~10}`. Altri esempi di questa procedura si trovano nel codice riprodotto nel paragrafo 5.3.

2.3.25 Direzione delle aste del basso

PMX rivolge automaticamente verso l'alto le aste delle note che si trovano sulla terza linea del pentagramma inferiore per lasciar posto alla cifratura. Per disattivare questa funzione e fare in modo che le aste di tali note vengano, come di consueto, rivolte verso il basso, basta aggiungere **B** all'inizio del *corpo* del codice.

³⁴Cfr paragrafo 2.3.8.

2.3.26 Macro

Una macro³⁵ è un solo simbolo che rappresenta una serie di simboli. Si tratta di un comando utile soprattutto quando vi sono parti da ripetere nella partitura. Un file **PMX** può contenere fino a 20 macro. Il comando **MR n** (dove n è il numero della macro, da 1 a 20) inizia la registrazione di una macro, mentre **M** la conclude. Tutto ciò che si trova fra questi due simboli oltre ad essere utilizzato normalmente dal programma, viene anche registrato per essere utilizzato in altri punti del file mediante il comando **MP n** (*macro play*). Se si desidera registrare una macro che non sia utilizzata subito dal programma basta usare al posto del comando **MR n** il simbolo **MS n** (*macro save*).

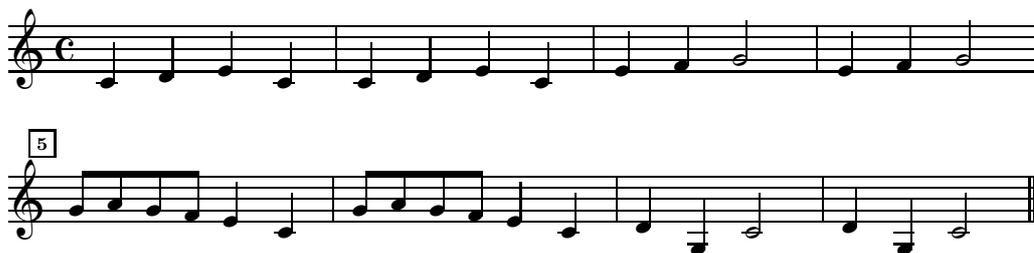


FIGURA 2.45 – Macro

La musica di figura 2.45 (*Fra Martino*), in cui ogni battuta si ripete due volte, ci permette di sperimentare l'uso di questi comandi. Eccone il codice.

```
1 1 4 4 0 6 0 0
1 2 20 0.0

t
./
Ab
It64
w160m
% Batt 1 (macro registrata e usata)
MR1 c4 d e c M
% Batt 2 (macro usata)
MP1 /
% Macro immagazzinata ma non usata
MS2 e4 f g2 M
% Batt 3 (macro usata)
MP2 /
% Batt 4 (macro usata)
MP2 /
% Batt 5 (macro registrata e usata)
MR3 g84 a g f e4 c M /
% Batt 6 (macro usata)
```

³⁵Per le macro in generale si veda il paragrafo 4.1.3.

```

MP3 /
% Batt 7 (macro registrata e usata)
MR4 d4 g- c2 M /
% Batt 8 (macro usata)
MP4 /
%
```

2.3.27 Comandi **T_EX** in linea

Non bisogna dimenticare che **PMX** è un preprocessore per **MusiX_{T_EX}**, il cui scopo è quello di semplificare la codificazione di una partitura. Come si è visto è possibile effettuare stampe raffinatissime anche senza conoscere nulla né di **T_EX** né di **MusiX_{T_EX}**. D'altro canto va ricordato che **MusiX_{T_EX}** offre funzioni che non sono ancora disponibili in **PMX**, come la notazione antica per il canto gregoriano oppure le librerie di simboli per le percussioni e per particolari strumenti musicali. **T_EX** poi mette a disposizione un'infinità di simboli tipografici e numerosissime funzioni grafiche. Conoscere almeno i fondamentali di **T_EX** e **MusiX_{T_EX}** non è insomma indispensabile, ma è comunque assai utile. **PMX** è un programma in divenire: è assai migliorato dalla sua prima apparizione ad oggi e continua a migliorare grazie alla costante dedizione del suo autore, Don Simons. Tuttavia, anche ciò che è già disponibile in **MusiX_{T_EX}**, ma non lo è ancora in **PMX**, può essere utilizzato mediante l'uso dei comandi **T_EX** in linea. In questo modo tutte le risorse **T_EX** e **MusiX_{T_EX}** sono utilizzabili all'interno di **PMX**.

Ci sono quattro modi per inserire comandi **T_EX** in linea in un codice `.pmx`. In tutti i casi il programma copia i comandi così come sono nel file `.tex` risultante. I primi tre tipi differiscono solo per il punto in cui i comandi sono copiati nel file `.tex`.

A questo punto è necessaria una brevissima parentesi su **MusiX_{T_EX}**. Il codice `.tex` di una partitura codificata con **MusiX_{T_EX}** contiene parti scritte nel puro linguaggio di **T_EX** (intestazioni, definizioni per semplificare l'introduzione di alcuni elementi della partitura ecc.), poi contiene il comando `\startmuflex`, che segnala l'apertura della parte strettamente musicale³⁶, quindi il comando `\startpiece` che predisporre i pentagrammi e le altre regolazioni valide per l'intera partitura, infine accetta le note vere e proprie poste fra i due comandi `\notes` ed `\enotes`. Alla fine della partitura vanno chiusi i comandi che erano stati aperti: `\endmuflex`, `\endpiece`.

A seconda del tipo di comando **T_EX** in linea che si intende inserire, è necessario scegliere attentamente il punto del file `.tex` in cui tale comando verrà inserito. È questa la ragione per cui ci sono quattro diversi tipi di comandi **T_EX** in linea. Il

³⁶Più esattamente segnala che le informazioni relative alla giustificazioni della pagina vanno immagazzinate nel file `nome.mx1`, utile per l'elaborazione in tre passi della pagina (*cfr* paragrafo 1.3.1).

quarto tipo permette l'uso di un codice $\text{T}_{\text{E}}\text{X}$ di più linee, mentre i primi tre consentono solo l'introduzione di una stringa di codice $\text{T}_{\text{E}}\text{X}$ di lunghezza non superiore a 128 caratteri, che cominci sulla prima colonna e che si concluda con \backslash . I tre tipi differiscono per il loro inizio: \backslash (tipo 1), $\backslash\backslash$ (tipo 2), $\backslash\backslash\backslash$ (tipo 3).

1. $\backslash \dots \backslash$ Il comando verrà trascritto nel file `.tex` immediatamente prima di quello relativo alla nota davanti alla quale è posto (all'interno dunque della coppia `\notes ... \enotes`)
2. $\backslash\backslash \dots \backslash$ Il comando verrà trascritto all'inizio del file `.tex`, immediatamente prima di `\startmuflex`
3. $\backslash\backslash\backslash \dots \backslash$ Il comando verrà trascritto all'inizio del *blocco* corrente, immediatamente prima di `\alaligne`, oppure di `\xbar`³⁷

Il quarto tipo di comando $\text{T}_{\text{E}}\text{X}$ in linea permette di trascrivere diverse linee di testo all'inizio del file `.tex`. Le linee devono trovarsi obbligatoriamente all'inizio del file `.pmx` e devono essere comprese all'interno di altre due linee contenenti il segno `---` (tre lineette), posto sulle prime tre colonne, cioè:

```
---
  comandi  $\text{T}_{\text{E}}\text{X}$  di più linee
---
```

Per ulteriori dettagli sui comandi $\text{T}_{\text{E}}\text{X}$ in linea e più in generale sul linguaggio di **MuSiX $\text{T}_{\text{E}}\text{X}$** si rinvia al manuale [3]. Qui basterà fare qualche esempio dell'uso di questi comandi. Un caso in realtà lo abbiamo già visto. Per utilizzare il pacchetto delle legature postscript *tipo M* è necessario usare un comando $\text{T}_{\text{E}}\text{X}$ in linea tipo 2, cioè $\backslash\backslash\text{input musixpss}\text{relax}\backslash$.

Per aggiungere simboli non previsti direttamente da **PMX**, come, ad esempio, il segno di pedale, si deve utilizzare un comando $\text{T}_{\text{E}}\text{X}$ in linea del tipo: $\backslash\text{zcharnote}\{-n\}\{\backslash\text{PED}\}\backslash$, dove n è un numero intero, con segno $+$ o $-$, che regola la posizione verticale del simbolo. Il segno di pedale di figura 2.46, per esempio, è stato ottenuto con il comando: $\backslash\text{zcharnote}-2\{\backslash\text{PED}\}\backslash$.



FIGURA 2.46 – Comandi $\text{T}_{\text{E}}\text{X}$ in linea (tipo 1)

Un comando in linea, $\backslash\text{nobarnumbers}\backslash$, che impedisce la stampa dei numeri di battuta, ci è già capitato di incontrarlo a pagina 43.

³⁷Due comandi che **PMX** introduce all'inizio di ogni blocco, poiché indicano rispettivamente l'«a capo» forzato o la forzata continuazione della riga. Sono comandi che si trovano fuori dalla coppia `\notes ... \enotes`.

Un altro esempio in proposito. Per impostazione predefinita **PMX** pone i numeri di battuta in un riquadro all'inizio di ogni sistema, come si vede in figura 2.43. Se vogliamo eliminare il riquadro e usare lo stile antico per i numeri (`\oldstyle`) è sufficiente premettere al codice trascritto a pagina 64 il seguente comando `\TeX` in linea (tipo 4):

```

---
\let\:=\relax\input musixtex\:\sepbarrules\input pmx
\def\writebarno{\ifnum\barno>1\lrlap{\oldstyle\the\barno\barnoadd}\fi}%
\def\shiftbarno{3.5\Interligne}
---
```

La figura 2.47 mostra l'effetto di questa modifica.

FIGURA 2.47 – Comandi `\TeX` in linea (tipo 4)

In questa occasione il comando `\sepbarrules` è ininfluenza, perché è presente un solo pentagramma, ma esso serve a separare le linee di battuta per ciascuno strumento anziché mantenerle continue per tutti i pentagrammi del sistema, secondo il modo predefinito da **MuSiX_{TeX}**. Un esempio dell'effetto di questo comando lo si può vedere nell'esempio riprodotto nel paragrafo 5.4.

Mediante il seguente comando si ottiene invece la numerazione ogni 5 battute.

```

---
\let\:=\relax\input musixtex\:\sepbarrules\input pmx
\nobarnumbers
\def\freqbarno{5}%
---
```

Altri esempi di comandi `\TeX` in linea si trovano nei codici riprodotti nel paragrafo 5.

2.3.28 Creazione del file midi

PMX può produrre file midi, i quali, se da un lato risultano, com'è ovvio, un po' troppo meccanici all'ascolto, dall'altro sono utilissimi per la correzione degli errori. L'introduzione di I all'inizio del primo blocco fa in modo che venga prodotto, oltre al file `.tex` anche un file `.mid`. I può trovarsi anche all'inizio di uno dei blocchi successivi (quando è necessario variare alcuni aspetti dell'esecuzione, come ad esempio il tempo) e può essere seguito da una serie di opzioni³⁸.

pmx	strumento	midi	pmx	strumento	midi	pmx	strumento	midi
pi	Grand Piano	1	va	Viola	42	a1	Sax Alto	66
rh	Rhodes Piano	5	vc	Violoncello	43	te	Sax Tenore	67
ha	Cembalo	7	cb	Contrabb.	44	bs	Sax Barit.	68
ct	Clavinet	8	vo	Voce	55	ob	Oboe	69
ma	Marimba	13	tr	Tromba	57	ba	Fagotto	71
or	Organo a canne	20	tb	Trombone	58	c1	Clarinetto	72
gu	Chitarra acust.	25	tu	Tuba	59	f1	Flauto	74
ab	Basso	33	fr	Corno franc.	61	re	Recorder	75
v1	Violino	41	so	Sax Soprano	65			

TABELLA 2.5 – Simboli midi

- `tx`: regola il tempo a x impulsi al minuto (predefinito: 96);
- `px`: inserisce una pausa di x quarti di nota;
- `i1i2...in`: specifica il numero di strumento midi secondo lo standard General Midi. $i1, i2, \dots, in$ sono numeri interi fra 1 e 128 o abbreviazioni di 2 lettere (*cfr* tabella 2.5). I numeri in sequenza vanno separati dai due punti (:). Questa opzione va inserita una sola volta nel file e il numero degli strumenti non può essere cambiato (predefinito: 7 –cembalo–).
- `vi1:i2...:in`: specifica la ‘velocity’ (cioè il volume) di ciascuno strumento (i due punti sono necessari). Il valore varia da 1 a 127 (predefinito: 127);
- `bi1:i2...:in`: specifica il bilanciamento stereo (i due punti sono necessari). Il valore varia da 1 a 128. 64 rappresenta la posizione centrata (predefinito), i valori inferiori potenziano il canale di sinistra e i valori superiori quello di destra;
- `MRi`: inizia la registrazione della macro numero i che sarà presente *solo* nel file midi. Il numero di macro i deve essere compreso fra 1 e 20. È utile per i ritornelli e i *da capo*.
- `M`: conclude la registrazione della macro.

³⁸Se ne vedano alcuni esempi nei codici nel paragrafo 5.

- **MP*i***: esegue la macro numero *i*. **Attenzione**: può essere attiva solo una macro alla volta e di un solo tipo (registrazione **MR** o esecuzione**MP**).
- **g*i***: stabilisce la pausa tra una nota e l'altra in midi *tic* (predefinito: 10, cioè 2/3 di una semibiscroma);
- **T+, -, i1 +, -, i2... +, -, in**: stabilisce il grado di trasposizione in semitoni (*i* deve essere multiplo di 12).

Lo standard General Midi assegna un numero da 1 a 128 a ciascuno degli strumenti predefiniti, divisi in famiglie, come ci mostra la tabella 2.6. La tabella 2.5 presenta invece i simboli abbreviati di due lettere previsti da **PMX**.

Non sono riconosciuti: ornamenti, appoggiature, acciaccature, volta, ritornelli, segni. Le legature sono effettuate solo se indicate con **s** o **(**, senza numero di identificazione.

2.3.29 Testo lirico

PMX non prevede l'inserimento del testo lirico, che deve quindi o essere posto mediante l'uso di comandi **T_EX** in linea e il pacchetto **Musixlyr** di Rainer Dunker, oppure mediante **M-Tx**, di cui parleremo nel capitolo 3.

2.4 Limiti

Per ogni file di codice sono fissati dei limiti. Alcuni sono sotto il diretto controllo dell'operatore, come il numero di pagine o di battute da inserire, altri riguardano elementi calcolati automaticamente dal programma e sono quindi fuori dal controllo diretto di chi utilizza il programma. Se uno o più di tali limiti viene superato **PMX** si blocca e stampa un messaggio di errore. Per evitare che ciò avvenga è consigliabile dividere su più file **.pmx** le partiture particolarmente lunghe³⁹.

2.4.1 Limiti sotto il diretto controllo dell'operatore

- 128 caratteri per linea di input.
- 12 pentagrammi.
- 2 linee di musica per pentagramma.
- 12 linee di musica per sistema.
- 125 sistemi.

³⁹Chi è esperto e sa compilare il programma dal sorgente fortran, può aumentare le capacità del programma. *Cfr* per questo il manuale di **PMX** a pagina 24 e quello di **M-Tx** a pagina 22.

- 600 battute.
- 40 ‘a capo’ forzati.
- 10 ‘a capo pagina’ forzati.

1-8	PIANO	9-16	CHROMAT. PERC.	17-24	ORGAN
1	Acoustic Grand Piano	9	Celesta	17	Drawbar Organ
2	Bright Acoustic Piano	10	Glockenspiel	18	Percussive Organ
3	Electric Grand Piano	11	Music Box	19	Rock Organ
4	Honky-tonk Piano	12	Vibraphone	20	Church Organ
5	Electric Piano 1	13	Marimba	21	Reed Organ
6	Electric Piano 2	14	Xylophone	22	Accordian
7	Harpichord	15	Tubular Bells	23	Harmonica
8	Clavinet	16	Dulcimer	24	Tango Organ
25-32	GUITAR	33-40	BASS	41-48	STRINGS
25	Acoustic Guitar (nylon)	33	Acoustic Bass	41	Violin
26	Acoustic Guitar (steel)	34	Electric Bass (finger)	42	Viola
27	Electric Guitar (jazz)	35	Electric Bass (pick)	43	Cello
28	Electric Guitar (clean)	36	Fretless Bass	44	Contrabass
29	Electric Guitar (muted)	37	Slap Bass 1	45	Tremelo Strings
30	Overdriven Guitar	38	Slap Bass 2	46	Pizzicato Strings
31	Distortion Guitar	39	Synth Bass 1	47	Orchestral Strings
32	Guitar Harmonics	40	Synth Bass 2	48	Timpani
49-56	ENSEMBLE	57-64	BRASS	65-72	REED
49	String Ensemble 1	57	Trumpet	65	Soprano Sax
50	String Ensemble 2	58	Trombone	66	Alto Sax
51	Synth Strings 1	59	Tuba	67	Tenor Sax
52	Synth Strings 2	60	Muted Trumpet	68	Baritone Sax
53	Choir Aahs	61	French Horn	69	Oboe
54	Voice Oohs	62	Brass Section	70	English Horn
55	Synth Voice	63	Synth Brass 1	71	Bassoon
56	Orchestra Hit	64	Synth Brass 2	72	Clarinet
73-80	PIPE	81-88	SYNTH LEAD	89-96	SYNTH PAD
73	Piccolo	81	Lead 1 (square)	89	Pad 1 (new age)
74	Flute	82	Lead 2 (sawtooth)	90	Pad 2 (warm)
75	Recorder	82	Lead 3 (calliope)	91	Pad 3 (polysynth)
76	Pan Flute	83	Lead 4 (chiff)	92	Pad 4 (choir)
77	Blown Bottle	84	Lead 5 (charang)	93	Pad 5 (bowed)
78	Shakuhachi	85	Lead 6 (voice)	94	Pad 6 (metallic)
79	Whistle	86	Lead 7 (fifths)	95	Pad 7 (halo)
80	Ocarina	87	Lead 8 (bass+lead)	96	Pad 8 (sweep)
97-104	SYNTH EFFECTS	105-112	ETHNIC	113-124	PERCUSSIVE
97	FX 1 (rain)	105	Sitar	113	Tinkle Bell
98	FX 2 (soundtrack)	106	Banjo	114	Agogo
99	FX 3 (crystal)	107	Shamisen	115	Steel Drums
100	FX 4 (atmosphere)	108	Koto	116	Woodblock
101	FX 5 (brightness)	109	Kalimba	117	Taiko Drum
102	FX 6 (goblins)	110	Bagpipe	118	Melodic Drum
103	FX 7 (echoes)	111	Fiddle	119	Synth Drum
104	FX 8 (sci-fi)	112	Shanai	120	Reverse Cymbal
121-128	SOUND EFFECTS				
121	Guitar Fret Noise				
122	Breath Noise				
123	Seashore				
124	Bird Tweet				
125	Telephone Ring				
126	Helicopter				
127	Applause				
128	Gunshot				

TABELLA 2.6 – Strumenti General Midi

- 18 cambi di tonalità.
- 20 pagine.
- 200 note per blocco di input.
- 15 battute per blocco di input.
- 101 legature per blocco di input.
- 74 cifre di basso per blocco di input.
- 37 gruppi di appoggiature o acciaccature per blocco di input.
- 74 appoggiature o acciaccature per ogni gruppo per blocco di input.
- 52 comandi \TeX in linea per blocco di input.
- 6 volta per blocco di input.
- 18 trilli per blocco di input.
- 62 note di accordo (inizianti per **z**) per blocco di input.
- 8 aste raggruppate per linea di musica per battuta.
- 40 raggruppamenti forzati di aste per linea di musica per blocco di input.
- 10 cambi di chiave per linea di musica per blocco di input.
- 24 note per raggruppamento di aste.
- 41 note per raggruppamento irregolare.

2.4.2 Limiti fuori dal diretto controllo dell'operatore

- 2000 gruppi di `\notes`.
- 20 gruppi di `\notes` per battuta.
- 20 spazi anti-collisione (eccetto gruppi irregolari e fine di battuta) per battuta.
- 20 spazi anti-collisione all'interno dei gruppi irregolari per battuta.
- 19 spazi anti-collisione rigidi in fine di battuta per sistema.
- 83 spazi anti-collisione rigidi in fine di battuta.
- 400 spazi standard anti-collisione per sistema.

- 100 spazi anti-collisione all'interno di gruppi irregolari per sistema.
- 1000 spazi standard anti-collisione.
- 200 spazi anti-collisione all'interno di gruppi irregolari.
- 24576 byte di dati MIDI prodotti per linea di musica.

2.5 Estrazione delle parti

Il programma **scor2prt**, distribuito insieme a **PMX**, permette l'estrazione automatica delle parti. Facendolo girare seguito dal nome del file `.pmx` senza estensione produce una serie di file `.pmx` al cui nome viene aggiunto un numero progressivo, che indica lo strumento (dal basso verso l'alto nel sistema). Un esempio. Abbiamo il file `quart.pmx` contenente la partitura di un quartetto con i pentagrammi disposti nel seguente modo (i numeri tra parentesi indicano l'ordine delle linee musicali nel file `.pmx`):

- violino I (4)
- violino II (3)
- viola (2)
- basso (1)

Per estrarre le parti usiamo il seguente comando:

```
scor2prt quart
```

Si otterranno in questo modo quattro nuovi file (`quart1.pmx`, `quart2.pmx`, `quart3.pmx`, `quart4.pmx`) in cui sono contenute rispettivamente le parti di basso, viola, violino II e violino I. A questo punto basta processare ciascun file con **PMX** per ottenere la stampa delle parti.

Ovviamente l'impaginazione delle parti e quella della partitura sono assai differenti, perciò è necessario aver cura di usare, *già nel codice sorgente* `.pmx`, opportuni accorgimenti. **PMX** prevede una serie di comandi rivolti esclusivamente a **scor2prt**, grazie ai quali tutte le informazioni relative all'estrazione delle parti sono inserite nel codice `.pmx` sorgente. Poiché i comandi relativi alle parti vengono inseriti tutti nel file sorgente non è necessario effettuare nessun altro intervento sui file prodotti da **scor2prt**, inoltre ogni variazione apportata in seguito nel file sorgente `.pmx` sarà automaticamente copiata esattamente nelle parti, facendo semplicemente girare il programma di estrazione.

Come si è visto il simbolo `%` introduce una linea di commento. Normalmente tutte le linee che iniziano con questo segno vengono ricopiate in tutte le parti. Ma

se una linea nelle prime due colonne presenta i simboli `%%`, allora quella linea e la successiva non sono ricopiate nelle parti. Se all'inizio di una delle linee ignorate si trova `h`, `l`, `Ti` o `Tc`, allora le linee ignorate saranno 3 anziché due.

Ogni linea che inizia con `%!` viene copiata esattamente nelle parti senza i due caratteri iniziali. Si tratta di una funzione utile per aggiungere comandi in tutte le parti staccate che non devono essere presenti nella partitura.

È anche possibile inserire un comando in una particolare parte e fare in modo che sia ignorato in tutte le altre. Per far ciò basta inserire nel file sorgente una linea che inizi con `%h`, dove `h` è il numero esadecimale (`1,2,...,9,a,b,c`). Per esempio, se vogliamo ottenere un 'a capo' forzato alla linea 15 a pagina 2 solo nella parte 11, bisogna aggiungere nel punto in cui si vuole andare a capo del file `.pmx` sorgente `%bL15P2` (dove `b` è il numero esadecimale che rappresenta l'undicesima parte del sistema dal basso).

Ci sono comandi che, sebbene siano permessi, come s'è visto, solo nella prima voce del blocco, sono copiati in tutte le parti (a meno che la linea precedente non contenga `%%` nelle prime due colonne), eccoli: `m`, `V`, `R`, `A`, `h`, `w`, `K`.

I comandi \TeX in linea tipo 2-4 sono copiati in tutte le parti, mentre il tipo 1 viene posto, come è ovvio, solo nella parte a cui appartiene.

Gli spazi orizzontali forzati (`X` senza `:`) non vengono copiati in alcuna parte. Se si desidera che, al contrario, ciò avvenga, si devono usare le opzioni già viste⁴⁰ per il comando `X`: `B` aggiunge lo spazio sia nelle parti che nella partitura, `P` aggiunge lo spazio solo nelle parti. Gli spazi aggiuntivi che riguardano un solo elemento o una sola voce vengono normalmente copiati solo nella parte in cui si trovano.

Il numero di sistemi di ciascuna parte è quello stabilito per la partitura. Se nelle parti si desidera avere un numero diverso di sistemi, bisogna usare il comando `Si` (dove `i` rappresenta il numero di sistemi totali) all'inizio del primo blocco, preceduto da `%!` (che copia il comando in tutte le parti), oppure da `%k` (che lo copia solo nella parte numero `k`). Con lo stesso comando e nello stesso modo è possibile stabilire un diverso numero di pagine aggiungendo `Pi` e una diversa grandezza del pentagramma (nelle parti il valore predefinito è 20) aggiungendo `mi`. Per esempio se voglio che la parte 4 abbia 20 sistemi, con pentagrammi di grandezza 16, disposti su due pagine, devo inserire all'inizio del primo blocco del file `.pmx` sorgente il comando `%4S20P2m16`.

Il simbolo `P` viene ignorato nella creazione delle parti, a meno che non sia preceduto da `%!` o da `%i`, secondo l'uso già visto⁴¹.

Il comando midi `I` non viene copiato nelle parti, a meno che non sia introdotto nella forma `%!` o `%i`.

Quando nella partitura ci sono diverse battute di pausa consecutive in una parte `scor2prt` le raggruppa in una sola battuta contenente una pausa multipla (`rm`).

⁴⁰ *cf* il paragrafo 2.3.19.

⁴¹ Si noti che `P` come opzione di `S` regola il numero totale di pagine in una parte, come opzione di `L`, forza un 'a capo' pagina e come comando a sé stante inizia la numerazione delle pagine.

Vedremo presto che anche quando si usa **M-Tx** l'estrazione delle parti viene fatta da **scor2prt**: si dovrà compilare il file `.mtx` e poi estrarre le parti dal file `.pmx` risultante. Attenzione però: il comando `@` che serve per la regolazione dell'altezza del testo lirico produce errori durante l'estrazione delle parti. Un altro errore è prodotto dai comandi `TeX` in linea necessari per modificare la grandezza e i margini della pagina. In questi e in eventuali simili casi è necessario localizzare e sostituire o eliminare il comando che risulta incompatibile con **scor2prt**. Un sistema per aggirare i due problemi qui indicati verrà fornito nel paragrafo [4.1.5](#).

Capitolo 3

Breve guida all'uso di M-Tx

M-Tx è un preprocessore per **PMX**, cioè legge un file `.mtx` e lo traduce in uno `.pmx`. Il suo autore, Dirk Laurie, si è proposto di rendere ancora più semplice il già agevole processo di introduzione della partitura previsto da **PMX** e in più permettere una altrettanto agevole introduzione del testo lirico. Per ottenere il primo dei due scopi egli ha semplificato e reso più intuitivo il *preambolo*¹, ha invertito l'ordine delle voci in maniera che, come avviene in partitura, la più acuta fosse in alto e la più grave in basso, ha previsto una serie di comandi abbreviati per l'introduzione degli accordi e dei segni di dinamica. Per ottenere il secondo scopo ha ideato un sistema che non ha eguali, in quanto a semplicità, in nessun altro programma di stampa che io conosca e che funziona in pratica come preprocessore per **Musixlyr**². A mio parere è bene usare **PMX** per la musica strumentale e **M-Tx** per quella vocale, per questo lo spazio dedicato in questa guida al primo dei due programmi è assai maggiore di quello dedicato al secondo.

Vediamo per prima cosa un confronto fra i due sistemi di codificazione. Se vogliamo ottenere la musica di figura 2.1 (pagina 28), anziché con **PMX**, con **M-Tx**, il codice da introdurre è il seguente:

```
Style: Quartet
Meter: C
Flats: 1
Pages: 1
Systems: 1
Size: 20
Name: {\rm{V1~I}} {\rm{V1~II}} {\rm{Va}} {\rm{Vc}}

%%Ab
%%It92ivcvavlvl
%%w170m
% Batt 1
```

¹ Vedi il confronto proposto a pagina 3.

² È anche possibile, ovviamente, usare le funzioni di **Musixlyr** direttamente in **PMX**: *cfr* il manuale di **Musixlyr**, pagina 15 e sgg.

```

%%h
%%Allegro
a2 Df d- |
r8 a- Df o. a o. a o. r a o. a o. a o. |
f8 Df o. f o. f o. f o. r f o. f o. f o. |
d8- Df o. d o. d o. d o. r d o. d o. d o. |

%Batt 2
e2 a- |
r8 b o. b o. b o. r c s o. c o. c o. |
r8 g o. g o. g o. r g o. g o. g o. |
r8 d o. d o. d o. r e o. e o. e o. |

% Batt 3-4
d8 s c s d e s g s f e s a | d4- zd+ cn1 s b a g s s a4 o. r |
d8 s c s d e s e s d c s a1+ s g s | f s g f e s d8 o. d o. cs4 r |
a2 t ad4 t cs8- | d8 s e f d s e4 o. r |
f8 s e f c s s d4 a | b2 s a4 s o. r |

```

Le somiglianze fra i due sistemi di codificazione (si veda il codice `.pmx` riprodotto a pagina 27) sono evidenti:

- Il *corpo* presenta, come s'è detto, le voci invertite, ma è sostanzialmente uguale a quello di **PMX**
- le righe che iniziano con `%%` contengono comandi **PMX** in linea. Di conseguenza, per introdurre un comando \TeX in linea lo si deve fare mediante un comando **PMX** in linea (`%%\`, `%%\ o` `%%\ \`).

Il *corpo* di **M-Tx** differisce da quello di **PMX**, oltre che per l'invertita disposizione delle voci, per la mancanza del segnale di fine voce (/) e per pochi altri particolari³. Il *preambolo* invece è assai diverso, poiché **M-Tx** esprime attraverso stringhe di testo ciò che **PMX** rappresenta mediante numeri.

3.1 Preambolo

Innanzitutto occorre ricordare che il *preambolo* di **M-Tx** esprime in maniera diversa gli stessi elementi presenti nel *preambolo* di **PMX**, dunque ci interessa qui sapere come dichiarare i valori necessari e non tanto quali siano tali valori.

I comandi del preambolo predefiniti di **M-Tx** sono i seguenti:

Ad eccezione di `Space`, `Start`, `PMX`, `Options`, `Enable`, `Disable` e `Octave`, si tratta dei comandi già visti in **PMX**, il cui uso risulta evidente dagli esempi.

³Fra cui l'impossibilità di usare la codificazione abbreviata dei valori puntati (*cf*r pagina 36) presente nel codice **PMX** alla battuta 3 nella parte della viola, ma evitata nel codice **M-Tx**.

Part: Recorder	Nome della parte (giustificato a sinistra, sopra il titolo)
Title: Clarinet Quintet	Titolo del brano.
Composer: Mozart	Nome del compositore (giustificato a destra, sotto il titolo)
Poet: Rellstab	Nome del poeta (giustificato a sinistra, sotto il titolo)
Meter: C/	<i>Alla breve</i> metro: notazione alternativa a 2/2.
Flats: 3	Armatura con 3 bemolli.
Sharps: 2	Armatura con 3 diesis.
Space: 6 0 3	Spazio aggiunto sotto i pentagrammi.
PMX: h10i	Comando PMX nel preambolo. funzione resa obsoleta da %.
Options: x	Usa l'opzione x durante la compilazione.
Pages: 2	Distribuisce il brano su due pagine.
Systems: 11	Distribuisce il brano su 11 sistemi.
Bars/line: 4	Pone 4 battute per linea.
Size: 16	Grandezza del pentagramma: 16 punti. Se non specificato: 20.
Style: Singer Piano	Sistema di tre pentagrammi per canto e piano.
Name: Dietrich Gerald	Nomi degli strumenti, degli esecutori etc.
Indent: 0.10	Rientro del primo sistema del 10% della lunghezza totale.
Start: @+1;@-3	Pone gli elementi specificati all'inizio delle linee delle voci.
Enable: doLyrics	Considera i paragrafi lirici.
Disable: doLyrics	Ignora i paragrafi lirici.
Octave: 4 4 3 3	Specifica l'ottava di ciascuna voce.

TABELLA 3.1 – Comandi del preambolo di **M-Tx**

Dell'opzione **PMX** non parleremo, visto che è più comodo usare per lo stesso scopo **%**. Occupiamoci ora degli altri comandi.

Space: *i1 i2 ... in* aggiunge *i* interlinea di spazio aggiuntivo fra i pentagrammi. Se ad esempio in una partitura per quartetto vogliamo inserire uno spazio aggiuntivo fra un sistema e l'altro (quindi sotto il pentagramma più basso) scriveremo: **Space:** 0 0 0 1. Se invece vogliamo aggiungere due spazi in uno spartito per canto e piano sotto il rigo del canto per far posto al testo, scriveremo **Space:** 2 0 0 (o anche solo **Space:** 2). Questo valore può essere modificato ponendo ulteriori comandi **Space** all'inizio di qualsiasi blocco.

Options: *x* permette di specificare le opzioni da usare durante la compilazione del file. Quando si lancia il programma, infatti, è possibile specificare una serie, anche complicata di opzioni⁴. Per non appesantire la linea di comando da digitare è possibile specificare le opzioni di compilazione direttamente nel file. Per esempio se desideriamo che sia stampata solo la musica, senza le parole, di una partitura e che siano evitati i messaggi di errore pedanteschi, specificheremo nel preambolo: **Options:** -mw.

Dalla versione 0.55 **M-Tx** permette un più semplice ed efficace controllo delle opzioni di compilazione mediante i comandi **Enable** e **Disable**, che servono, rispettivamente, per abilitare e disabilitare la funzione selezionata. Tutti i comandi

⁴ Vedi paragrafo 3.4.

che seguono possono essere selezionati o disattivati in tal modo. Ecco la lista dei comandi che sono **attivati** in modo predefinito:

splitShortcut divide i comandi abbreviati 2:1 e 3:1 in due e codifica esplicitamente la loro durata

newWordShortcut permette l'inserimento dei comandi abbreviati 2:1 e 3:1 come due simboli separati

multiFile considera le linee contrassegnate da **Include**:

doChords considera le linee contrassegnate da **C**:

doUptext considera le linee contrassegnate da **U**:

doLyrics considera le linee contrassegnate da **L**: e gli altri controlli delle liriche

unbeamVocal evita di raggruppare le note delle linee vocali

hideBlindSlurs Nasconde {~ e (~

interpretSlurs trascrive come **s** e **t** le legature

uptextOnRests sincronizza i simboli **uptext** non solo con le note ma anche con le pause

Ecco la lista dei comandi **disabilitati** in modo predefinito

solfaNoteNames usa come nomi di note **d r m f s l t** (do re mi fa sol la ti) anziché i consueti nomi inglesi; i simboli che coincidono con le note, come **r** che significa sia re che pausa vanno fatti precedere da "

pedanticWarnings sopprime gli avvisi di compilazione pedanteschi

ignoreErrors ignora gli errori e compila sino al termine

instrumentNames Predispone il rientro del primo sistema e stampa il nome dello strumento predefinito

beVerbose stampa ogni avviso possibile di compilazione

debugMode stampa i messaggi di compilazione che possono essere utili per trovare gli errori nel codice

L'opzione **Size**: definisce la grandezza del pentagramma. **M-Tx** consente una scelta più ampia rispetto a **PMX**: 13, 16, 20, 24 o 29 punti⁵.

Start: serve per porre il comando di regolazione dell'altezza del testo lirico (@), anziché direttamente nella voce a cui si riferisce, nel preambolo. Per l'uso del comando @ si veda pagina 93.

Importantissima è l'opzione **Style**, che consente di definire il numero e il tipo dei pentagrammi del sistema. Esistono diversi *stili* predefiniti. Eccoli:

⁵ Cfr il manuale di **M-Tx** a pag. 24.

SATB: Voices S,A T,B; Choral; Clefs G F
 SATB4: Voices S A T B; Choral; Clefs G G G8 F
 SINGER: Voices S; Vocal; Clefs G
 PIANO: Voices RH LH; Continuo; Clefs G F
 ORGAN: Voices RH LH Ped; Continuo; Clefs G F F
 SOLO: Voices V; Clefs G
 DUET: Voices V1 Vc; Clefs G F
 TRIO: Voices V1 Va Vc; Clefs G C F
 QUARTET: Voices V1 V2 Va Vc; Clefs G G C F
 QUINTET: Voices V1 V2 Va Vc1 Vc2; Clefs G G C F F
 SEXTET: Voices V1 V2 Va1 Va2 Vc1 Vc2; Clefs G G C C F F
 SEPTET: Voices V1 V2 Va1 Va2 Vc1 Vc2 Cb; Clefs G G C C F F F

Se, ad esempio, dobbiamo stampare un brano per piano, basterà specificare nel preambolo `Style: Piano` per avere un sistema di due pentagrammi, uniti dalla graffa, con le chiavi di violino e basso. Attenzione: se il brano inizia per qualche motivo con una chiave diversa, come avviene nell'esempio di pagina 3, è necessario ridefinire lo stile.

Per la musica vocale sono stati predisposti due diversi tipi di sistema:

- `SATB: Voices S,A T,B; Choral; Clefs G F`
- `SATB4: Voices S A T B; Choral; Clefs G G G8 F`

Il primo pone le quattro voci su due soli pentagrammi, recanti le chiavi di violino e basso, mentre il secondo impiega quattro pentagrammi, con le chiavi (dall'alto in basso) di violino, violino, violino tenorizzata, basso.

Se gli stili predefiniti non sono soddisfacenti è possibile crearne di nuovi, specificando: *nome dello stile, voci, tipo di sistema e chiavi*. Ecco, per esempio, come si ottiene un rigo di piano contenente quattro linee di musica, anziché due.

`Piano4v: Voices MD1,MD2 MS1,MS2; Continuo; Clefs G F`

Poi, per indicare lo stile appena creato, basta scrivere: `Style: Piano4v`.

Il *nome* può essere scelto a piacere. I tipi di voce vanno specificati mediante simboli⁶. Se i simboli sono separati l'uno dall'altro da spazi, allora ciascuna voce si trova su un pentagramma, mentre se due simboli sono separati da una virgola (senza spazi) allora le due voci si trovano sullo stesso pentagramma. Sono disponibili quattro diversi tipi di sistema: `Continuo`, `Vocal`, `Choral` e `Group`. `Continuo` fa in modo che i pentagrammi appartenenti a uno stesso strumento (due nel caso del piano, tre nel caso dell'organo, ad esempio) siano uniti da una graffa. `Vocal` imposta un rigo adatto al canto, in cui, cioè, le aste delle note non vengono automaticamente unite. `Choral` è pensato per la musica corale: i pentagrammi sono uniti

⁶Attenzione a non scegliere simboli già impiegati dal programma per altri comandi, come `L`, `U`, `C`, `1` o `L1`.

e le aste delle note sono lasciate separate. **Group** indica che i pentagrammi appartengono a una famiglia strumentale e sono perciò uniti, ma, contrariamente al caso precedente le aste delle note sono, come di norma, raggruppate automaticamente.

Per specificare le chiavi si possono usare i simboli **PMX** (*vedi* pagina 32) oppure **F** (basso), **C** (contralto), **G** (violino) e **G8** (violino tenorizzato).

3.1.1 Omissione dei segni di battuta

Nella musica antica talvolta non sono usati i segni di battuta. **M-Tx** permette l'introduzione di righe musicali senza linee di battuta indicando nel preambolo ad esempio **Meter: 0/4**. Se il numeratore è 0 il programma considera ogni pentagramma come una sola battuta e ne calcola automaticamente il valore (*vedi* l'esempio a pagina 98). Come denominatore va indicato il valore più piccolo fra quelli presenti.

3.2 Corpo

M-Tx usa in pratica lo stesso linguaggio di **PMX**, perciò in questo paragrafo ci occuperemo soprattutto delle differenze fra i due programmi e daremo per assodate le funzioni già note. Va ricordato intanto che diversi comandi (soprattutto quelli che si trovano su linee a sé, come **A**, non accettati direttamente da **M-Tx**, possono essere introdotti come comandi **PMX** in linea mediante **%**).

3.2.1 Misure incomplete

M-Tx riconosce automaticamente le misure incomplete poste all'inizio (*levare*) o alla fine di un brano, a patto che siano segnalate dalla stanghetta di battuta **|**. Nel caso del *levare* il corretto valore viene aggiunto al *preambolo* del file **.pmx** risultante; se si tratta della battuta conclusiva il programma calcola automaticamente il cambio di metro invisibile da inserire nel codice **.pmx**.

3.2.2 Linee di battuta

Il sistema di codificazione⁷ è assai intuitivo e semplice:

- | Linea di battuta normale
-] Doppia stanghetta di fine movimento (sottile-grossa)
- || Doppia stanghetta
- :| Ritornello a sinistra
- |: Ritornello a destra
- :|: Ritornello doppio a destra e a sinistra

⁷Ricavato da ABC2MT_EX di Chris Walshaw.

- Le linee di battuta normali sono, come in **PMX**, opzionali e servono per facilitare il calcolo dei valori e l'individuazione di eventuali errori, a meno che (qui sta la differenza) non indichino misure incomplete all'inizio o alla fine del brano (*cfr* paragrafo 3.2.1).
- Le doppie stanghette e i ritornelli devono essere posti nella voce più bassa (la principale in **PMX**), ma per chiarezza è consigliabile metterli anche nelle altre. I segni in ogni caso appariranno, ovviamente, in tutti i pentagrammi.
- Le linee di battuta normali devono comparire solo in corrispondenza della fine della misura (altrimenti **PMX** stampa un messaggio di errore), mentre le doppie stanghette e i ritornelli possono apparire anche in mezzo alla battuta.

3.2.3 Numero di voci per pentagramma

PMX permette di variare il numero di voci (1 o 2) per ciascun pentagramma ad ogni blocco, mediante il segno *//*⁸. Anche **M-Tx** lo consente, ma solo occasionalmente e con alcune restrizioni:

- nello stile deve essere prevista solo una voce per il pentagramma in questione
- la voce deve terminare con la stanghetta (|)
- le due voci da inserire nel pentagramma devono essere scritte una dopo l'altra, separate dal segno *//* *sulla stessa linea*
- si deve usare questa funzione solo per passi molto brevi

3.2.4 Cambio di metro

Il cambio di metro deve essere posto all'inizio di una battuta (senza **%**), dopo una battuta completa nel metro precedentemente in uso e *in tutte le voci*. Può essere indicato mediante una frazione, come nel preambolo (3/4, 4/4 ecc.), oppure attraverso uno dei due modi già visti per **PMX** (*cfr* paragrafo 2.3.13). Il metodo di codifica di **PMX** è obbligatorio nei casi particolari in cui non c'è corrispondenza fra la frazione specificata e il metro stampato (come nel tempo *a cappella*). Eventuali mutamenti di tonalità, indicazioni di *volta* ecc. vanno posti *dopo* il nuovo segno di metro **M-Tx!metro**. Infine, va ricordato che la prima battuta dopo un *levare* non può contenere cambi di metro.

3.2.5 Gruppi ritmici (travature)

Negli stili strumentali (es.: **Style: piano**) le aste delle note sono raggruppate automaticamente, mentre in quelli vocali (es.: **Style: SATB**) non viene effettuato

⁸*cfr* pagina 33.

alcun raggruppamento. In questi casi per ottenere i gruppi ritmici è necessario usare le parentesi quadre ([]) (Vedi paragrafo 2.3.7).

3.2.6 Legature

Vengono usate le stesse regole viste per **PMX** (cfr paragrafo 2.3.6). È inoltre prevista la funzione di ‘continuazione’ (da usarsi quando sulla stessa nota finisce una legatura di valore e ne inizia un'altra.

Quando una legatura è aperta all'interno di un'altra legatura, come previsto da **PMX**, si possono usare due volte i segni di inizio e fine (es.: (... (...)).

Per le legature di valore si usano le parentesi graffe {} anziché quelle tonde; in tal caso il segno di continuazione è }{.

3.2.7 Accordi

Valgono tutte le regole di **PMX** (cfr paragrafo 2.3.3). Inoltre **M-Tx** consente di porre le note aggiunte a ciascun accordo (quelle che in **PMX** iniziano per **z**) in una linea separata. Ne diamo solo un esempio e rimandiamo il lettore al manuale di **M-Tx**⁹. Ecco il codice **M-Tx** necessario per ottenere la musica riprodotta in figura 2.16.

```
Style: Piano
Meter: m2400
Flats: 1

%%Ab
%%w150m
%1-5
c2 | df | b | bn | es |
C: etfra gtbd t dtertgn engs gscs
f2- | gf | ef | dn | cs+ |
C: acf bdtgt gnbet bn+en esgscs
```

Si noti che:

- gli accordi vanno inseriti nella linea, iniziante con **C:**, immediatamente sottostante quella della voce a cui si riferiscono
- la direzione dell'asta dell'accordo è quella della nota principale
- contrariamente al solito, ma *solo* nella riga degli accordi si deve usare **l** per la testa spostata a sinistra (anziché **e**) e **t** per il bemolle (anziché **f**).

⁹Pagine 10 e sgg.

3.2.8 Segni di dinamica e di espressione

Per questo genere di segni **M-Tx** possiede un efficiente sistema di codifica, alternativo e precedente a quello offerto da **PMX**¹⁰. Vediamone le caratteristiche principali.

- I segni di dinamica e di espressione, se non si modificano le impostazioni predefinite, vengono stampati sopra il pentagramma.
- I simboli vanno introdotti in una linea a sé, iniziante con **U**: (*uptext*), posta sopra il pentagramma a cui si riferiscono.
- **M-Tx** riconosce automaticamente i simboli musicali e li distingue da quelli testuali. Se il simbolo inserito è un segno di dinamica convenzionale (*f*, *mf*, *p*, *rfz*, ecc.) **M-Tx** lo stampa usando le fonti di **MusiXTeX** adatte, se invece è una normale stringa di testo, allora usa il carattere tondo predefinito (*cmr*).
- Per modificare il carattere della stringa di testo da stampare si deve inserire, prima del simbolo e a distanza di uno spazio, il comando **!it** per il corsivo e **!bf** per il grassetto.
- La stringa di testo non deve contenere spazi. Nel caso che ciò sia necessario si deve usare `~` anziché lo spazio.
- Ciascun simbolo va separato dal successivo mediante uno spazio. Ogni carattere presente nella linea che inizia con **U**: è considerato dal programma come un simbolo (di dinamica o testuale) da porre in corrispondenza di una nota. Il primo carattere sarà posto in corrispondenza della prima nota, il secondo in corrispondenza della seconda e così via. Il carattere `~` indica che in corrispondenza della nota su cui è posto non vanno inseriti simboli.

Le battute iniziali della parte del flauto del *Prélude à l'après midi d'un faune* di Debussy, riprodotte in figura 3.1, serviranno da esempio. Ecco il codice **M-Tx** impiegato.

```
Style: Solo
Meter: 9/8
Sharps: 4
Size: 20
Pages: 1
Systems: 2
Space: 10
Name: Flute
```

¹⁰**PMX** ha introdotto un proprio sistema di codifica delle dinamiche solo dalla versione 2.3. Cfr paragrafo 2.3.8.

Très modéré

Flute

p *doux et expressif*

FIGURA 3.1 – Segni di dinamica e di espressione in M-Tx

```

%%Ab
%%B
%%It44ifl
%%w160m
%%\nobarnumbers\
%%\input musixpss\relax\
%1
%%h
%% Tr\'es mod\'er\'e
U: @v-7 p !it doux~et~expressif
%%%%%%%% Le due righe seguenti vanno poste sulla stessa riga!
cd4+ sA sBt+0+.5 [l c8 sBt sCt c1 sCt b ][ as8x3 an g ]
[l gnd8 a1 b bs sA+0+0+6 ] |

%2
%%%%%%%% Le due righe seguenti vanno poste sulla stessa riga!
cd4 sA sBt [l c8 sBt+0+.5 sCt c1 sCt b ][ as8x3 an g ]
[l gnd8 a1 b bs sA+0+0+6 ] |

%3-4
U: @+5 <12 ~ ~ ~ ~ >6
[ c8 sA d g ] e4,g- bd4 sBt+0+.5 | b8 sBt b c as4 sA+5+0+7 r8 r4 r8 ||
%
```

M-Tx consente di regolare la posizione dei simboli e permette anche di introdurre le forcelle del crescendo e gli accordi per chitarra. Ecco in sintesi le caratteristiche di questi comandi.

- **Aggiustamenti verticali.** Si ottengono mediante il simbolo @¹¹.

¹¹Contrariamente a quanto accade per il testo lirico, il simbolo @ va posto nella riga che inizia con U: e non in quella che contiene i simboli musicali.

- @v sistema i simboli che seguono sotto il pentagramma anziché sopra
 - @^ sistema i simboli che seguono sopra il pentagramma
 - @+n o @-n sistema i simboli che seguono a $n\backslash\text{internote}$ sopra (+) o sotto (-) il punto predefinito¹²
 - @=n sistema i simboli che seguono all'altezza precisa di $n\backslash\text{internote}$ sopra (o sotto) il pentagramma
- **Aggiustamenti orizzontali.** Si ottengono con @> e @<. @< modifica l'allineamento dei simboli che seguono di una posizione verso sinistra, mentre @> fa il medesimo verso destra. Per impostazione predefinita ciascun simbolo viene allineato da **M-Tx** a destra della nota. Per ottenere l'allineamento al centro si deve inserire una volta il segno @<, per ottenere l'allineamento a sinistra si deve introdurre il medesimo simbolo due volte¹³. Per tornare all'allineamento centrato o a destra sarà sufficiente introdurre, rispettivamente una o due volte, il segno @>.
 - **Accordi per chitarra.** Mediante i comandi *uptext* è possibile inserire accordi per chitarra, stampati in formato testo. In questo caso per indicare il diesis si usa # e per il bemolle %.
 - **Crescendo e diminuendo.** I simboli < e > segnalano il punto in cui, rispettivamente, la forcilla del crescendo e quella del diminuendo iniziano, mentre i simboli <. e >. indicano il punto in cui tali forcelle finiscono. Se invece si inserisce un solo simbolo del tipo <n o >n, allora verrà inserita, a partire dalla posizione del simbolo, una forcilla di crescendo o di decrescendo della lunghezza di $n\backslash\text{elemskip}$.
 - **Restrizioni**
 - Non è possibile inserire più di un simbolo per nota. Nel caso che sia necessario porre, per esempio, una forcilla di crescendo insieme a un segno di dinamica è però possibile associare il comando *uptext* con i comandi D di **PMX** e, ovviamente, con altri comandi $\text{T}_{\text{E}}\text{X}$ in linea.
 - Di norma non è consentito inserire più di una forcilla di crescendo contemporaneamente in voci diverse. In questo caso è possibile usare i comandi **PMX**, ma, *attenzione!*, D< e D> non sono utilizzabili se la forcilla si estende per più di una battuta¹⁴.

¹²Si tratta di comandi *relativi* e non assoluti, dunque la posizione definita dal comando si aggiunge a quella stabilita precedentemente. Se ad esempio nella linea precedente era stato inserito il comando @+5 e nella corrente viene inserito @+2, la posizione sarà complessivamente di $7\backslash\text{internote}$ più in alto rispetto a quella predefinita.

¹³Eventuali altri spostamenti a sinistra saranno ignorati dal programma.

¹⁴È condizione essenziale per il crescendo e il decrescendo in **PMX** che i segni di inizio e fine simbolo siano posti nello stesso blocco, ma **M-Tx** trascrive ogni battuta in un blocco a sé, rendendo così inutilizzabile il comando.

- Per creare la forcella del crescendo, **M-Tx** inserisce nel codice `.pmx` dei comandi \TeX in linea, che, sfortunatamente, sono incompatibili con la sintassi impiegata dal pacchetto postscript *tipo K*. In questo caso dunque si deve compilare il file usando le fonti bitmap di **MusiX \TeX** , oppure, come abbiamo fatto nell'esempio di figura 3.1, il pacchetto postscript *tipo M*.

3.2.9 Macro

Anche **M-Tx** permette l'uso di macro (come quelle viste per **PMX**). Per farlo si devono introdurre nelle linee musicali gli stessi simboli già visti per **PMX**, cioè MRn , M e MPn (*cfr* paragrafo 2.3.26). I comandi, secondo l'impostazione predefinita, vengono semplicemente ricopiati nel file `.pmx`. Ciò può produrre dei problemi, visto che **M-Tx** non può tenere il conto delle battute inserite in questo modo. Sono però disponibili due diversi modi, per superare questa difficoltà. Basta introdurre *nel preambolo* uno dei due comandi **Enable** (*cfr* pag. 80) che seguono.

Il primo è **Enable: expandMacro**. In tal caso tutte le macro presenti saranno sviluppate e il risultante file `.pmx` non conterrà alcuna macro, ma le battute musicali saranno interamente ricopiate. In tal caso è possibile usare fino a 99 macro (non solo 20 come in **PMX**).

Il secondo è **Enable: countMacro**. In questo caso **M-Tx** ricopia nel file `.pmx` solo i segni di macro, ma tiene il conto delle battute. **countMacro** ha però una significativa restrizione: ciascuna macro (registrata o eseguita) non può superare l'ambito della battuta in cui è inserita.

3.3 Testo lirico

3.3.1 Testo inserito fra le linee del codice

M-Tx permette di introdurre con una facilità estrema il testo lirico, poiché funziona come preprocessore di **Musixlyr**. Il corale n. 40 della *Passione secondo Giovanni* di Bach, riprodotto in figura 3.2, ci offre un buon punto di partenza.

Il codice impiegato per stampare il corale è il seguente:

```
Style: SATB4
Meter: C
Sharps: 4
Size: 20
Pages: 1
Systems: 2
Name: {\it{Soprano}} {\it{Alto}} {\it{Tenore}} {\it{Basso}}

%%\font\rx = cmr8
%%\rx
```

The image displays a musical score for four vocal parts: Soprano, Alto, Tenore, and Basso. The score is in G major (one sharp) and common time (C). The lyrics are in Italian and describe the freedom of Christ through his sacrifice. The lyrics are as follows:

Soprano:
 Durch dein Gefäng-nis Got - tes Sohn, ist uns die Frei - heit kom - men,
 Dein Ker-ker ist der Gna - den-thron, die Freistatt al - ler From - men,

Alto:
 Durch dein Gefäng-nis Got - tes Sohn, ist uns die Frei - heit kom - men,
 Dein Ker-ker ist der Gna - den-thron, die Freistatt al - ler From - men,

Tenore:
 Durch dein Gefäng-nis Got - tes Sohn, ist uns die Frei - heit kom - men,
 Dein Ker-ker ist der Gna - den-thron, die Freistatt al - ler From - men,

Basso:
 Durch dein Gefäng-nis Got - tes Sohn, ist uns die Frei - heit kom - men,
 Dein Ker-ker ist der Gna - den-thron, die Frei-statt al - ler From - men,

The second system of the score continues the lyrics:

denn gingst du nicht die Knecht - schaft ein, müßt' uns - re Knechtschaft e - wig sein.

FIGURA 3.2 – J. S. Bach, *Passione secondo Giovanni*, n. 40

```

%%Abp
%%\nobarnumbers\
%%It72ibatuclobb58:70:60:68T+0-12+0+0
%%w160m
%%h200m
%0-2
@+1 [ e8 f ] | g4 a b b | a g f of b |
L: Durch dein Ge-f\ang-nis Got-tes Sohn, ist
L: Dein Ker-ker ist der Gna-den-thron, die

```

b4- | e e e e | [e8 d] e4 d of f |
 L: Durch dein Ge-f\"ang-nis Got-tes Sohn, ist
 L: Dein Ker-ker ist der Gna-den-thron, die
 @+1 [g8 a] | b4 c dn [c8 b] | c4 b b of ds |
 L: Durch dein Ge-f\"ang-nis Got-tes Sohn, ist
 L: Dein Ker-ker ist der Gna-den-thron, die
 @-6 e4 | e c g c | f [g8 a] b4 of b |
 L: Durch dein Ge-f\"ang-nis Got-tes Sohn, ist
 L: Dein Ker-ker ist der Gna-den-thron, die

%3-4a

c d e [d8 c] | m3400 c2 b4 :|
 L: uns die Frei-heit kom-men,
 L: Frei-statt al-ler From-men,
 f as [b8 f] [f g] | m3400 (g4 as) f :|
 L: uns die Frei-heit kom-men,
 L: Frei-statt al-ler From-men,
 c f [b8- c] [d e] | m3400 ([e d] c4) d :|
 L: uns die Frei-heit kom-men,
 L: Frei-statt al-ler From-men,
 as [g8 f] [g a] b4 | m3400 (e- f) b- :|
 L: uns die Frei-heit kom-men,
 L: Frei-statt al-ler From-men,

%4b-6

mo400 @+2 b | m4400 e4 [d8 c] c4 b | a [g8 f] f4 of b |
 L: denn gingst du nicht die Knecht-schaft ein, m\"u{\ss}t'
 mo400 @+2 g | m4400 c4 b [b8 a] [a g] | [g f] e4 d of dn |
 L: denn gingst du nicht die Knecht-schaft ein, m\"u{\ss}t'
 mo400 @+2 e | m4400 g4 g [g8 f] [f d] | [e b] b4 b of [g8 f] |
 L: denn gingst du nicht die Knecht-schaft ein, m\"u{\ss}t'
 mo400 @+1 e | m4400 c4 g a b | [c8 d] e4 b of b |
 L: denn gingst du nicht die Knecht-schaft ein, m\"u{\ss}t'

%7-8

a g [f8 g] a4 | g f e of
 L: uns-re Knecht-schaft e-wig sein.
 cn b [a8 b] c4 | [b8 e] [e b] b4 of
 L: uns-re Knecht-schaft e-wig sein.
 e4 es f f | [g8.a] [b f] g4 of
 L: uns-re Knecht-schaft e-wig sein.
 cn cs dn ds | e b e of
 L: uns-re Knecht-schaft e-wig sein.
 %

Il sistema di codificazione resta quello visto sin qui¹⁵. Per il testo lirico si devono notare invece le seguenti caratteristiche.

- Le liriche si ottengono introducendo il testo preceduto da L: nella riga sottostante la linea musicale a cui si riferiscono.
- Le sillabe di una parola sono separate da -.
- Se alla fine di una parola vi è un vocalizzo **M-Tx**¹⁶ stampa automaticamente una linea continua della lunghezza necessaria; se il vocalizzo si trova in una sillaba all'interno della parola allora la linea è tratteggiata.
- Il simbolo @ (seguito da un numero intero con segno) serve a regolare l'altezza delle linee del testo.
- Per i caratteri particolari (es.: ß) si usa, come al solito, la codifica T_EX.
- Se si intende modificare il carattere tipografico predefinito si deve usare la codifica T_EX.
- Quando le aste sono unite **M-Tx** riconosce che si tratta di un vocalizzo e dispone solo una sillaba sotto la prima nota.
- Lo stesso accade con le legature, ma attenzione: solo i simboli () { } sono riconosciuti, mentre s e t funzionano come legature musicali e non come indicatori dei vocalizzi.

Chiunque conosca i sistemi WYSIWYG, come **Finale**, comprenderà che avere sullo schermo tutto il testo in un semplice file ascii è enormemente più comodo che aprire e chiudere almeno una finestra di dialogo per ogni sillaba che si intende introdurre o modificare!

3.3.2 Testo inserito in un paragrafo a parte

È anche possibile (ed è il sistema che personalmente preferisco) riunire il testo in un solo paragrafo lirico e assegnarlo poi alle parti tramite il comando L:. Ecco la stessa musica (quella di figura 3.2) con il testo inserito in un paragrafo a parte.

```
Style: SATB4
Meter: C
Sharps: 4
Size: 20
Pages: 1
```

¹⁵Si noti in particolare l'uso dei comandi T_EX in linea, introdotti come comandi **PMX** in linea (`%\`) e fra essi un comando midi abbastanza complicato per fare in modo che la parte del tenore sia trasposta all'ottava bassa e che si produca l'effetto stereofonico attraverso il bilanciamento.

¹⁶A volere essere esatti è **Musixlyr** a farlo.

```

Systems: 2
%Options: s
Name: {\it{Soprano}} {\it{Alto}} {\it{Tenore}} {\it{Basso}}

%%\font\rviii = cmr8
%%\rviii

{sopA}={altoA}={tenA}={bassA}
Durch dein Ge-f\ang-nis Got-tes Sohn,
ist uns die Frei-heit kom-men,

{sopB}={altoB}={tenB}={bassB}
denn gingst du nicht die Knecht-schaft ein,
m\u{\ss}t' uns-re Knecht-schaft e-wig sein.

{sopC}={altoC}={tenC}={bassC}
Dein Ker-ker ist der Gna-den-thron,
die Frei-statt al-ler From-men,

%%Ab
%%\nobarnumbers\
%%It72ibatuclobb58:70:60:68T+0-12+0+0
%%w160m
%%h200m
%0-2
@+1 [ e8 f ] | g4 a b b | a g f of b |
L: {sopA,sopC}
b4- | e e e e | [ e8 d ] e4 d of f |
L: {altoA,altoC}
@+1 [ g8 a ] | b4 c dn [ c8 b ] | c4 b b of ds |
L: {tenA,tenC}
@-6 e4 | e c g c | f [ g8 a ] b4 of b |
L: {bassA,bassC}

%3-4a
c d e [ d8 c ] | m3400 c2 b4 :|
f as [ b8 f ] [ f g ] | m3400 ( g4 as ) f :|
c f [ b8- c ] [ d e ] | m3400 ( [ e d ] c4 ) d :|
as [ g8 f ] [ g a ] b4 | m3400 ( e- f ) b- :|

%4b-6
mo400 @+2 b | m4400 e4 [ d8 c ] c4 b | a [ g8 f ] f4 of b |
L:{sopB}
mo400 @+2 g | m4400 c4 b [ b8 a ] [ a g ] | [ g f ] e4 d of dn |
L:{altoB}
mo400 @+2 e | m4400 g4 g [ g8 f ] [ f d ] | [ e b ] b4 b of [ g8 f ] |

```

```
L:{tenB}
mo400 @+1 e | m4400 c4 g a b | [ c8 d ] e4 b of b |
L:{bassB}
```

```
%7-8
a g [ f8 g ] a4 | g f e of
cn b [ a8 b ] c4 | [ b8 e ] [ e b ] b4 of
e4 es f f | [ g8.a ] [ b f ] g4 of
cn cs dn ds | e b e of
```

In questo caso non è stato più necessario ricopiare lo stesso testo sotto ciascuna voce e il codice appare più ordinato, inoltre le eventuali correzioni risulteranno semplificate, poiché non sarà necessario effettuarle (almeno in casi come questo) quattro volte. Il procedimento è il seguente:

- scrivere, dopo il preambolo, il testo sillabato e le eventuali definizioni del carattere tipografico, preceduto da un'etichetta che lo identifichi (es.: `{sopA}`)
- etichettare il testo che in diverse voci è identico (ma, attenzione, identica deve essere anche la sillabazione) nel modo seguente:
`{sopA}={altoA}={tenA}={bassA}`
- assegnare il testo alla voce (o alle voci), ponendo, nel modo già visto, il comando `L:` nella riga sottostante la voce (o le voci) seguito dalla sola etichetta e non dal testo (es.: `L: {sopA}`)
- Nel caso che sotto le stesse note vadano poste due (o più) strofe diverse (come nelle battute 1–4 del nostro esempio) basta inserire le due etichette (es.: `{sopA,sopC}`)

Il sistema a paragrafo di testo permette anche di cambiare linea di testo a metà di una battuta a patto che non si debba iniziare una nuova lirica con `L:`, ma che il comando sia già stato dato. Un esempio. Poniamo che una parte lirica sia associata al paragrafo `{soprano}` e a metà di una battuta la si voglia associare con il paragrafo `{basso}`. Scrivendo `c4 d {basso}` e `f` si otterrà che dal `mi` in poi le parole stampate sotto quella parte saranno quelle del paragrafo `{basso}`.

3.3.3 Sinalefe e vocalizzi

Semplicissima è anche l'introduzione della *sinalefe*, che viene prodotta dal semplice pedice (es.: `so-lo_e pen-so-so`). Poche battute di una cantata di Bononcini ce ne forniscono un esempio (figura 3.3).

Il codice da cui la figura 3.3 è stata ricavata è il seguente.



FIGURA 3.3 – Sinalefe e vocalizzi

```
Style: Singer
Meter: 3/8
Flats: 2
Size: 20
Pages: 1
Systems: 1
```

```
%\font\rviii = cmr8
%\rviii
```

```
%%Ab
%%w140m
@+2 b8 [ b c1 d ] | a8 [ a b1 c ] | b8 [ b c1 d ] | c8 [ c d1 e ] |
L: Quan-do par-li_e quan-do ri-di
```

3.3.4 Melismi

L'esempio riprodotto in figura 3.4, (la parte iniziale dell'*Ave verum* di Josquin des Prés), ci fornisce una più ampia dimostrazione dell'impiego dei melismi. Può infatti accadere che sia necessario un melisma che prosegua su note che non sono legate, né raggruppate mediante le aste. In tali casi è necessario ricorrere a legature invisibili, cioè porre legature che iniziano con \sim o con $\{\sim$ e finiscono con $\}\sim$ o con $\}\sim^{17}$, come è stato fatto nel codice da cui è stata ricavata la figura 3.4.

```
Style: ST
ST: Voices S T; Choral; Clefs G G
Meter: C/
Size: 20
Pages: 1
Systems: 3
Space: 2 6
%Options: s
Name: {\it{Superius}} {\it{Tenor}}
```

```
%\def\writebarno{\ifnum\barno>1\lrlap{\oldstyle\the\barno\barnoadd}\fi}
%\def\shiftbarno{0\Interligne}
```

```
{sup}
```

¹⁷È anche possibile usare i comandi \TeX di **Musixlyr**, vedi pagina 100.

Superius

A - - ve ve - - rum — cor - pus na - tum,

Tenor

A - ve ve - rum, — ve - - rum cor - pus

9

cor - pus na - - tum ex Ma - ri - a — Vir - - - gi -

na - - - tum — ex Ma - - ri - -

17

ne, ex Ma - ri - a Vir - - - - gi - - ne. —

a Vir - - - - gi - ne. —

FIGURA 3.4 – Melismi

A-ve ve-rum
 cor-pus na-tum,
 cor-pus na-tum
 ex Ma-ri-a Vir-gi-ne,
 ex Ma-ri-a Vir-gi-ne.

{ten}
 A-ve ve-rum, ve-rum
 cor-pus na-tum
 ex Ma-ri-a Vir-gi-ne.

```
%%Ab
%%It142ituobb58:70
%%w160m
%1-4
@+2 rp | rp | c0+ | d2 {~ ( e |
L: {sup}
@+6 f0 | g2 {~ ( a | a4 ) f }~ ( f2 | f ) {~ ( c+ |
L: {ten}
```

%5-8

```
e4 ) c }~ ( c2 | c ) f | e ( d | d ) c |
c4 ) [ bf8 a ] a4.f }~ | f0 | r2 ( bf | b ) a |
```

%9-12

```
r2 f | e4 {~ d2 [ c8 bf ] | c2 }~ f- | r c+ |
{~ g2 f | g0 }~ | ( a | a ) |
```

%13-14

```
d f | {~ e4.d e }~ {~ ( d |
rp | g |
```

%15-16

```
[ d8 ) c d c ] d4 ( c | c ) [ bf8 a ] }~ g2 |
f | e |
```

%17-20

```
f2 f+ | e d | c {~ ( f | f ) e |
f0 | {~ g | a2.f | f g |
```

%21-25

```
d ( c | c4 ) bf a2 }~ | {~ gd4 f8 g2 }~ | ( f0 | f ) of ||
bf ( a | a4 ) g ( f2 | f4 ) [ e8 d ] }~ e2 | ( f0 | f ) of ||
```

3.3.5 Tipi particolari di legature e impiego di musixlit

Quan - tus tre - mor est fu - tu - rus, quan - do Ju - - dex est ven - tu - rus,

cun - cta stri - cte di - scus - su - rus. Tu - - ba mi - - rum

spar - - gens so - num, per se - pul - chra re - gi - o - num,

FIGURA 3.5 – Legature senza vocalizzi e impiego di **musixlit**

L'esempio riprodotto in figura 3.5 mostra alcuni altri comandi particolari. Eccone il codice¹⁸:

```

Style: Basso
Basso: Voices B; Vocal; Clefs F
Meter: 0/4
Size: 20
Pages: 1
Systems: 3
Space: 7
%%\input musixlit

{basso}
Quan-tus tre-mor est fu-tu-rus,
quan-do Ju-dex est ven-tu-rus,
cun-cta stri-cte di-scus-su-rus.
Tu-ba mi-rum spar-gens so-num,
per se-pul-chra re-gi-o-num,

%%Abp
%%B
%%It142iba
%%w160m
%%\nobarnumbers\
%%\let\ql=\cnql\let\hl=\cnhl\
%1
%% Le due righe seguenti vanno poste sulla stessa riga!
@+6 ((h f2 e4 f2 d4 e c1 d2 d4 )
      ((h f \nolyr\ g2 g4 \beginmel\ f2 e4 \endmel\ d c1 e f e2 d4 )
L: {basso}

%2
%% Le due righe seguenti vanno poste sulla stessa riga!
((hu a2l c4l d2 c4l e f e2 d4 )
      ((h \beginmel\ a+ \endmel\ c2 c4 \beginmel\ b2 g4 \endmel\ a2 a4 )

%3
((h \beginmel\ a2 g4 \endmel\ f1 g a2 d4- ) ((h f1 e1 f2 d4 e c1 d2 d4 ) Rb

```

- **Omissione delle linee di battuta.** Per inserire musica senza linee di battuta è necessario introdurre il metro 0/4 (come denominatore va scelto il valore più piccolo fra quelli presenti nella battuta).
- **Sillabazione in presenza di legature.** Le legature in questo esempio hanno il solo scopo di indicare la frase musicale, non il vocalizzo. Per fare in modo che **M-Tx** ponga le sillabe anche sotto le note legate bisogna raddoppiare il simbolo di inizio legatura: ((o {{.

¹⁸Modifico l'esempio seguendo le indicazioni presenti nella traduzione tedesca del presente manuale di Christof K. Biebricher.

- **Comandi di Musixlyr.** In questo caso per ottenere i vocalizzi sono stati introdotti due comandi $\text{T}_{\text{E}}\text{X}$ in linea, fra quelli previsti da **Musixlyr**. `\nolyr\` posto davanti a una nota fa in modo che quella nota sia saltata nella sillabazione; `\beginmel\ ... \endmel\` fanno in modo che le note comprese fra questi due comandi siano saltate nella sillabazione.
- **Uso di musixlit.** Attraverso dei comandi $\text{T}_{\text{E}}\text{X}$ in linea è anche possibile utilizzare l'estensione di **MusiX $\text{T}_{\text{E}}\text{X}$** denominata **musixlit**¹⁹, che permette l'impiego di un genere di notazione intermedio fra quello gregoriano e quello barocco. Per far ciò è necessario invocare questa estensione con il comando `%%\input musixlit`. Per stampare le note quadrate antiche al posto di quelle odierne bisogna usare inoltre il seguente comando $\text{T}_{\text{E}}\text{X}$:
`%%\let\ql=\cnql\let\hl=\cnhl\`²⁰

- **Definizione di comandi abbreviati.** Attraverso la definizione di nuovi comandi $\text{T}_{\text{E}}\text{X}$ è anche possibile abbreviare quelli troppo lunghi per rendere più agevole la scrittura del codice. I comandi `%%\let\BM=\beginmel\`, `%%\let\EM=\endmel\` e `%%\let\NL=\nolyr\` permettono di utilizzare `\BM\`, `\EM\` e `\NL\` al posto di `\beginmel\`, `\endmel\` e `\nolyr\`.

3.3.6 Controllo individuale dei versi; altri tipi di legatura

Se abbiamo due o più versi sotto la stessa linea di musica, può capitare che un verso richieda un vocalizzo e l'altro no. Il codice utilizzato per l'esempio di figura 3.6 ci mostra come procedere.



FIGURA 3.6 – Manipolazione individuale dei versi

Style: Singer
 Meter: C/
 Flats: 1
 Size: 20
 Pages: 1
 Systems: 1

{VI}

¹⁹Si veda il manuale [3], pp. 92–93.

²⁰Per ulteriori informazioni su questo ed altri comandi $\text{T}_{\text{E}}\text{X}$ si rimanda alle opere indicate in bibliografia, e in particolare [11], [17], [4] e [5] (traduzione italiana del precedente). Per altri esempi simili e per altre applicazioni di **musixlit**, si veda il paragrafo 3.3.7.

```

\‘E qua-si gior-no_or-mai

{VII}
Sal-go so-pra_un au-to-bus

%%Ab
%%B
%%It142iba
%%w100m
%%\nobarnumbers\
%Batt. 1
%%%% Le due righe seguenti vanno poste sulla stessa riga!
(" \verses{\beginmel}\ c8+ \verses{,\beginmel}\
      c4d )( \verses{\endmel}\verses{,\endmel}\ c ) c8 a | c d c2 Rz
L: {VI,VII}

```

In questo esempio si trovano alcuni comandi nuovi:

- **Legature tratteggiate.** Si ottengono iniziando il simbolo di legatura con (" o con {. L'effetto sulla sillabazione è uguale a quello che si ottiene con il raddoppio del primo segno (es.: ((), cioè la sillabazione prosegue anche sulle note legate, inoltre la legatura è tratteggiata²¹.
- **Segno di continuazione della legatura.** Quando sulla stessa nota finisce una legatura di valore e ne inizia un'altra è possibile inserire il segno di continuazione)(al posto del segno di fine legatura. In questo modo si evita di introdurre due segni diversi (di inizio e di fine) riferiti alla stessa nota.²²
- **Manipolazione individuale dei versi.** Usando le funzioni viste in precedenza se due strofe diverse si trovano sotto la stessa musica la sillabazione viene effettuata nello stesso modo. Per ottenere una sillabazione individuale per ciascuna strofa è necessario usare dei comandi \TeX in linea fra quelli previsti da **Musixlyr**. Basta includere il comando che ci occorre (es.: `\nolyr`) all'interno del comando `\verses{}`. Ad esempio, se vogliamo usare `\nolyr` solo nella prima di due strofe allora è sufficiente introdurre, nella linea musicale in questione il comando `\verses{\nolyr}`, mentre se vogliamo che lo stesso comando sia applicato solo alla seconda strofa, allora dobbiamo indicarla mediante l'aggiunta di una virgola: `\verses{,\nolyr}`. Una eventuale terza strofa sarebbe indicata mediante `\verses{,,}` e così via. La stessa cosa si può ottenere con i comandi `\beginmel ... \endmel`, come accade nel codice impiegato per l'esempio di figura 3.6.

²¹Si noti che se si usa il comando **PMX** per ottenere legature tratteggiate ((b), **M-Tx** tratterà la legatura come normale e non porrà sillabe sotto le note in essa comprese.

²²Si noti che in questo caso però le due legature sono considerate da **M-Tx** identiche per quanto riguarda la sillabazione, che quindi continua regolarmente.

3.3.7 Segni di battuta per la musica antica²³

L'estensione **musixlit**, di cui ci siamo già occupati nel paragrafo 3.3.5, permette di creare segni di battuta posti fra i pentagrammi anziché sul rigo. Per far ciò si devono usare i comandi in linea `%%\input musixlit` (per richiamare l'estensione **musixlit**) e `%%\interbarrules` (per fare in modo che i segni di battuta siano posti fra i pentagrammi). Nel caso in cui le battute siano di durata irregolare e non si voglia stampare alcuna indicazione di metro, si deve specificare nel *preambolo* un metro che abbia come numeratore 0 e come denominatore il valore più basso presente (es.: `Meter: 0/4`)²⁴. Quando il numeratore è 0 **M-Tx** calcola automaticamente il valore di ogni battuta e lo scrive nel file **PMX** risultante, con i necessari cambiamenti di tempo invisibili. In questo caso verrà posto un segno di battuta fra i pentagrammi alla fine di ogni blocco, quindi bisogna avere l'accortezza di scrivere una sola battuta di musica per ogni blocco. Spesso queste suddivisioni si ricavano implicitamente dalla stessa musica, come accade nell'esempio, tratto dal *Salterio di Ginevra*, proposto in figura 3.7 e ottenuto dal codice seguente²⁵.

```
Style: SATB
Flats: 0
Meter: 0/4
Pages: 1
Systems: 4
Space: 2 0
Size: 20
PMX: h180m w167m
%%\input musixlit
%%\interbarrules
%%\nobarnumbers
```

```
{ps42}
```

```
Ain-si qu'on oit le cerf brui-re Pour-chas-sant le frais des eaux,
Ain-si mon coeur qui sou-pi-re, Sei-gneur a-pr'`es tes ruis-seaux,
Va tous-jours cri-ant, sui-vant Le grand, le grand Dieu vi-vant;
H\'e-las don-ques, quand se-ra-ce Que ver-ray de Dieu la fa-ce?
```

```
c2+,b c,b a4 g f2 e r
@+1 g2,g g,g e4 e c2 c r
L: {ps42}
c2+,d e,d c4 b a2 g r
U: ~~~c.f.
Arb K-1-2 c2+,g c,g a4 e f2 c r
```

²³L'autore di questo paragrafo è **Christof K. Biebricher**. Ricavo il testo dall'edizione tedesca del manuale.

²⁴Cfr il paragrafo 3.1.1.

²⁵Si noti che la musica è stata trasposta un tono più in basso con il comando K-1-2.

Ain - si qu'on oit le cerf brui - re Pour - chassant le frais des eaux,

Ain - si mon coeur qui sou - pi - re, Sei - gneur a - près tes ruis - seaux,

Va toujours cri - ant, sui - vant Le grand, le grand Dieu vi - vant;

Hé - las don - ques, quand sera - ce Que ver - ray de Dieu la fa - ce?

FIGURA 3.7 – Segni di battuta fra i pentagrammi

g2, b c, c c2 b c r
 e2, g g, a g g g r
 c2, d e, f e d c r
 c2+, g c, f- c+ g c- r

c2, b c, b a4 g f2 e r
 g2, g g, g e4 e c2 c r
 c2, d e, d c4 b a2 g r
 c2+, g c, g a4 e f2 c r

g2, b c, c c2 b c r
 e2, g g, a g g g r
 c2, d e, f e d c r
 c2+, g c, f- c+ g c- r

```
c2,c c,c c4 b c2 r
g2,g e,a g4 g e2 r
e2,e g,f e4 d e2 r
c2+,c c,f- c4+ g c2 r
```

```
c2,c c,c c4 c b2 r
e2,e f,e a4 g g2 r
g2,g a,g f4 e d2 r
c2,c f-,c f4 c g2+ r
```

```
c2,d d,c a4 b c2 a r
g2,bf a,a e4 g g2 f r
e2,g f,e c4 d e2 c r
c2,g d+,a a4 g c2 f- r
```

```
c2,c a,g g4 f g2 g0 []
g2,g c-,c b4 c d2 e0 []
e2,e f,e d4 c2,b c0 []
c2,c f,c g4+ a g2 c0- []
```

Simili linee di battuta poste fra i pentagrammi si usano oggi spesso per rendere più leggibile a colpo d'occhio la musica antica, con la differenza rispetto alla notazione moderna, che la linea di battuta non rappresenta alcun confine per il valore delle note. Può capitare infatti che il valore di una nota si estenda oltre la linea di battuta. **M-Tx** e **PMX** però stampano un messaggio di errore e si bloccano quando una battuta è troppo lunga o troppo corta. Bisogna quindi usare un espediente, come quello impiegato nell'esempio 3.8, che propone un sistema alternativo di notazione dell'*Ave verum* di Josquin des Prés già visto nell'esempio 3.4. Eccone il codice.

```
Style: ST
ST: Voices S T; Choral; Clefs G G
Meter: C/
Size: 20
Pages: 1
Systems: 3
Space: 2 6
Name: {\it{Superius}} {\it{Tenor}}

%%\input musixlit
%%\def\writebarno{\ifnum\barno>1\lrlap{\oldstyle\the\barno\barnoadd}\fi}
%%\def\shiftbarno{0\Interligne}
%%\def\WoHb{%
%%\def\hu##1{\let\hl\hlsav\let\hu\husav\wh{##1}}%
%%\def\hl##1{\let\hl\hlsav\let\hu\husav\wh{##1}}%
```

The image shows a musical score for two voices: Superius and Tenor. The Superius part is written on a single staff with a treble clef and a common time signature. The Tenor part is written on a single staff with a bass clef and a common time signature. The lyrics are written below the notes, with hyphens indicating syllables that span across multiple notes. The score is divided into three systems, each starting with a measure number (1, 9, and 17). The lyrics are: "A - - ve ve - rum cor - pus na - tum," "A - ve ve - rum, ve - - rum cor - pus na - - tum ex Ma - ri - a Vir - - gi - na - - tum ex Ma - - ri - - ne, ex Ma - ri - a Vir - - gi - - ne." and "a Vir - - gi - - ne."

FIGURA 3.8 – Uso dei segni di battuta nella musica antica

```

%%\def\PoHb{\def\hl##1{\let\hl\hlsav\hlp{##1}}%
%%\def\hu##1{\let\hu\husav\hup{##1}}%
%%\def\BoWb{\def\wh##1{\let\wh\whsav\breve{##1}}}%
%%\def\HoQb{\def\qu##1{\let\qu\qusav\hu{##1}}%
%%\def\ql##1{\let\ql\qlsav\hl{##1}}}%
%%\interbarrules

```

```

{sup}
A-ve ve-rum
cor-pus na-tum,
cor-pus na-tum
ex Ma-ri-a Vir-gi-ne,
ex Ma-ri-a Vir-gi-ne.

```

```

{ten}
A-ve ve-rum, ve-rum
cor-pus na-tum
ex Ma-ri-a Vir-gi-ne.

```

```

%%w160m

```

```

%1-2
@+2 rp | rp |
L: {sup}
@+6 Abp f0 | g2 \PoHb\ {~ a |
L: {ten}

%3-4
c0+ | d2 {~ \PoHb\ e |
r4b f }~ \WoHb\ f2 | r2b {~ \PoHb\ c+ |

%5-8
r4b c }~ \WoHb\ c2 | r2b f | e \WoHb\ d | rb c |
r4b [ bf8 a ] a4.f }~ | f0 | r2 \WoHb\ bf | rb a |

%9-12
r2 f | e4 {~ d2 [ c8 bf ] | c2 }~ f- | r c+ |
{~ g2 f | g0 }~ | \BoWb\ a | r0b |

%13-14
d f | {~ e4.d e }~ {~ ( d |
rp | g |

%15-16
[ d8 ) c d c ] d4 \HoQb\ c | r4b [ bf8 a ] }~ g2 |
f | e |

%17-20
f2 f+ | e d | c {~ \WoHb\ f | rb e |
f0 | {~ g | a2.f | f g |

%21-25
d \PoHb\ c | r4b bf a2 }~ | {~ gd4 f8 g2 }~ | \BoWb\ f0 of ||
bf \PoHb\ a | r4b g \PoHb\ f2 | r4b [ e8 d ] }~ e2 | \BoWb\ f0 of ||

```

Si tratta di utilizzare valori delle note che non superino la durata della battuta e poi di modificare l'aspetto delle note (secondo la codifica prevista da **MusiX_TE_X**) in maniera da ottenere quello corrispondente al valore sovrabbondante²⁶

La serie di definizioni che segue il preambolo serve (oltre a variare il carattere dei numeri di battuta nel modo già visto) per creare alcuni comandi fondamentali:

- `\WoHb\` stampa una semibreve anziché una minima
- `\PoHb\` stampa una minima puntata anziché senza punto

²⁶Il tipo di comandi impiegati in questo caso implica una conoscenza non superficiale sia di **T_EX** che di **MusiX_TE_X**. I meno esperti possono utilizzare i comandi di questo esempio anche solo in maniera operativa e trovare ulteriori informazioni nel paragrafo 4.1.3 e nella bibliografia là indicata.

- `\BoWb\` stampa una breve anziché una semibreve
- `\HoQb\` stampa una minima anziché una semiminima

Nel codice si inseriscono, come si è già detto, valori corrispondenti alla battuta (in questo caso corrispondenti ad una semibreve). Quando è necessario stampare un valore sovrabbondante si farà precedere la nota dal comando adatto fra quelli definiti sopra. Infine si completerà la battuta seguente introducendo una pausa invisibile. Per esempio, la linea del *tenor* della battuta 2 dell'*Ave verum* risulta sovrabbondante. Per ottenerla è stato introdotto il valore di due minime (`g2 a`), poi è stato modificato l'aspetto della seconda minima per aggiungervi il punto mediante il comando `\PoHb\`, infine è stata completata la battuta successiva con una pausa di semiminima. Ecco insomma come vanno scritte le battute 2 e 3 (omettiamo le legature):

```
g2 \PoHb\ a | r4b f f2 |
```

3.3.8 Due linee vocali sullo stesso pentagramma

È possibile stampare il corale bachiano della *Passione secondo Giovanni*, già visto nell'esempio 3.2, su due pentagrammi anziché su quattro (come si vede nell'esempio 3.9) ponendo due voci su ogni pentagramma. **M-Tx** mette a disposizione per questo scopo lo stile **SATB**. Il codice va in questo caso modificato come segue:

```
Style: SATB
```

```
Meter: C
```

```
Sharps: 4
```

```
Size: 20
```

```
Pages: 1
```

```
Systems: 2
```

```
Space: 10
```

```
Name: {\it{S/A}} {\it{T/B}}
```

```
{pA}
```

```
Durch dein Ge-f"ang-nis, Got-tes Sohn,  
ist uns die Frei-heit kom-men, ~
```

```
{pB}
```

```
Dein Ker-ker ist der Gna-den-thron,  
die Frei-statt al-ler From-men, denn
```

```
{pC}
```

```
gingst du nicht die Knecht-schaft ein,  
m\u{ss}t' uns-re Knecht-schaft e-wig sein.
```

```
%%w167m
```

%0-2

```
[ e8 f ] | g4 a b b | a g f of b |
L: {pA,pB}
b4- | e e e e | [ e8 d ] e4 d f |
[ g8 a ] | b4 c dn [ c8 b ] | c4 b b ds |
Abp e4 | e c g c | f [ g8 a ] b4 ofd b |
```

%3-4

```
c d e [ d8 c ] | c2 b4 :| b |
f as [ b8 f ] [ f g ] | g4 as f :| g |
c f [ b8- c ] [ d e ] | [ e d ] c4 d :| e |
as [ g8 f ] [ g a ] b4 | e- f b- :| e |
```

%5-6

```
e4 [ d8 c ] c4 b | a [ g8 f ] f4 of b |
L: {pC}
c4 b [ b8r a ] [ ar g ] | [ gr f ] e4 d dn |
g4 g [ g8 f ] [ f d ] | [ e b ] b4 b [ g8 f ] |
c4 g a b | [ c8 d ] e4 b ofd b |
```

%7-8

```
a g [ f8 g ] a4 | g f e of |]
cn b [ a8 b ] c4 | [ b8 e ] [ er b ] b4 |]
e4 es<.6 f f | [ g8.a ] [ b f ] g4 |]
cn cs>.15 dn ds | e b e ofd |]
```

Si osservi che, essendo la sillabazione identica in tutte e quattro le voci, si può scrivere il testo una sola volta fra i due pentagrammi, evitando le ripetizioni. Si è anche rinunciato in questo caso a far coincidere il segno di ritornello con la fine della linea. Questo permette di semplificare il codice, perché non c'è più bisogno dei cambi di metro invisibili né dell'«a capo» forzato.

3.3.9 Testo lirico sopra e sotto il pentagramma

A volte può essere necessario disporre due linee vocali sullo stesso pentagramma con le righe di testo poste una sopra e l'altra sotto il rigo, come avviene nella breve aria handeliana (*Serse*, II, 4) riprodotta in figura 3.10. Per far ciò è necessario introdurre la seconda voce non con il semplice comando L:, ma con *Lnomevoce:*, dove *nomevoce* è il nome della voce (specificato nel preambolo). Diamo uno sguardo al codice da cui è stata ricavata la figura 3.10.

```
Style: SA Basso
SA: Voices S,A; Choral; Clefs G
Basso: Voices Vc; Clefs F
Meter: 3/4
Sharps: 1
```

S/A

Durch dein Gefängnis, Got-tes Sohn, ist uns die Frei - heit kom - men,
Dein Kerker ist der Gna-denthron, die Freistatt al - ler From - men, denn

T/B

5

gingst du _ nicht die Knecht - schaft ein, müßt' uns - re Knechtschaft e - wig sein.

FIGURA 3.9 – Più voci sullo stesso pentagramma

Size: 20
 Pages: 1
 Systems: 3
 Space: 3 7

```
%%\def\writebarno{\ifnum\barno>1\lrlap{\oldstyle\the\barno\barnoadd}\fi}
%%\def\shiftbarno{0\Interligne}
```

{Romilda}

L'a-me-r{\`o}, L'a-me-r{\`o};
 l'em-pia sor-te vuol co-s{\`i},
 L'a-me-r{\`o}, L'a-me-r{\`o}, L'a-me-r{\`o}, L'a-me-r{\`o}.

{Serse}

L'a-me-re-te? L'a-me-re-te?
 E pur sem-pre vi tra-d{\`i},
 se ben fie-ro v'in-gan-n{\`o};
 L'a-me-re-te? L'a-me-re-te? L'a-me-re-te?

```
%%Ab
%%B
%%It112ibaob
%%w160m
```

L'ame - rò, L'ame - rò;
 L'ame - re - te? L'ame - re - te? E pur sempre vi tra-
 l'empia sor - te vuol co - sì, L'ame - rò,
 di, se ben fie - ro v'ingannò; L'ame-
 L'ame - rò, L'a-me - rò, L'ame - rò.
 re - te? L'ame - re - te? L'ame - re - te?

FIGURA 3.10 – Voci sopra e sotto il pentagramma

%1-5

```

@~+3 rp+5 | r4+7 r+7 f8+ g | e2 r4+5 | r+7 r+7 d8 e | c2 r4+6 |
L: {Romilda}
@v r4 r g8 e | b4+ b r | r r g8 e | b4+ b r | r-5 r-5 a8 b |
LA: {Serse}
e2,d | ds,r | e,r | gs,r | a,r |

```

%6-11

```

rp+6 | r4+7 r+7 b8.c | d.fn e4.b | c2,r+7 | rp | r4+7 r+7 b8 f |
c8.e c4.b | b2,r-6 | rp-7 | r4-5 r-5 c8 b | a4.g f8 e | b2+,r-6 |
f2,r | g,r | gs,r | a,r | c-,r | b,r |

```

%12-15

```

g2,r+7 | r4+8 r+8 d8+ a | b2,r+8 | r4+7 r+7 f8+.g |
r4-8 r-8 b8 g | d4+ d r | r-5 r-5 e8 a- | b4 b r |
e2,r | f,r | g,c 6 | b2 # r4 |

```

```
%16-18
e2,r+5 | r4+7 r+7 f8- g | ed2
r4 r g8 e | b4+ b- r-9 | rp-9
g2,c- | b,b | ed2-
```

Come si vede le voci disponibili (dall'alto in basso) sono: **S**, **A**, **Basso**. **S** è la voce principale del pentagramma ed **A** la voce ausiliaria. Il testo della voce principale è stato introdotto nella maniera consueta (L: {Romilda}), mentre per la voce ausiliaria è stato impiegato il comando **LA:{Serse}**, dove **A** è il nome della seconda voce specificato nel *preambolo*.

3.4 Opzioni di compilazione

Abbiamo visto²⁷ che per compilare il codice `.mtx` si usa il comando

```
prepmx nomefile
```

operazione che dà come risultato un file `.pmx`. **M-Tx** offre diverse opzioni di compilazione, che talvolta possono appesantire notevolmente la linea di comando, come avviene in questo caso estremo²⁸:

```
prepmx -bciw nomefile \partitur\texinput \partitur\mtx\miostile.txt
```

Per evitare di dover scrivere ogni volta un così lungo comando si può usare **Options** nel preambolo (*cfr* pagina 81). Un'altra più comoda soluzione, disponibile dalla versione 0.60, è l'uso di **Enable** o **Disable** (*cfr* in proposito il paragrafo 3.1).

Ecco la descrizione delle opzioni (i comandi **Enable**, **Disable** equivalenti sono posti fra parentesi).

- b Unisce le aste delle note di valore pari o inferiore alla croma, anche se lo stile è **Vocal** or **Choral** (**Disable** `unbeamVocal`).
- c stampa solo la linea musicale principale e ignora gli accordi (**Disable** `doChord`).
- D stampa i messaggi di compilazione utili per individuare gli errori nel codice (**Enable** `debugMode`).
- f Per i nomi delle note usa il sistema tonico sol-fa, cioè `d r m f s l t` per *do, re, mi, fa, sol, la, ti*. Il programma tradurrà tali nomi in quelli inglesi `c d e f g a b` nel file `.pmx` (**Enable** `solfaNoteNames`).

²⁷ *Cfr* paragrafo 1.3.3.

²⁸ In cui sono specificate la directory di scrittura del file, quella in cui si trova il file di stile `.txt`, oltre ad alcune altre opzioni.

- i Ignora gli errori (`Enable ignoreErrors`).
- m Stampa solo la musica e ignora i versi (`Disable doLyrics`).
- n Effettua il rientro del primo sistema e stampa i nomi degli strumenti predefiniti (`Enable instrumentNames`).
- t Ignora le linee 'uptext' (`Disable doUptext`).
- u Sincronizza l'uptext con la sola musica e non con le pause (`Disable uptextOnRests`).
- v Stampa gli avvisi in modo verboso (`Enable beVerbose`).
- w Stampa gli avvisi pedanteschi (`Enable pedanticWarnings`).
- 0, -1, ..., -9 Seleziona il caso specificato da `Case`²⁹.

²⁹ Cfr il manuale di M-Tx, pagina 14.

Capitolo 4

Approfondimenti e integrazioni

4.1 Comandi MusiX \TeX e \TeX

Lo studio della lingua di MusiX \TeX e di \TeX esula dall'intento di queste pagine. Il lettore può peraltro trovare le necessarie istruzioni nelle opere indicate in bibliografia. L'impiego di comandi \TeX all'interno di partiture codificate nella lingua di **PMX** o **M-Tx** offre però inesauribili risorse, che consentono di ottenere risultati non facilmente raggiungibili con altri sistemi di stampa musicale. Ne presentiamo alcuni esempi. Se ne possono trovare altri nella lista di discussione già citata [1], da cui in gran parte vengono anche i presenti.

4.1.1 Unione di più file

Poniamo di avere stampato con **PMX** una sonata per pianoforte in tre movimenti in tre file diversi: `son1.pmx`, `son2.pmx` e `son3.pmx`. Per unire insieme le tre diverse parti in modo da ottenere un solo file di stampa (`son.dvi` o `son.ps`), con la numerazione progressiva delle pagine, si deve fare in questo modo:

- Processare i tre file `.pmx` per ottenere i file `son1.tex`, `son2.tex` e `son3.tex`.
- Creare il file `son.tex`, che serve per unire i tre file con il seguente contenuto:

```
\input musixtex
\startmuflex
\let\startmuflex\empty
\let\endmuflexsav\endmuflex
\let\endmuflex\endinput
\input son1
\input son2
\input son3
\endmuflexsav
\bye
```

- Processare, nei tre passi consueti ($\text{T}_{\text{E}}\text{X} \implies \text{musixflx} \implies \text{T}_{\text{E}}\text{X}$) il file `son.tex`. Si otterrà il file `son.dvi` contenente l'intera partitura.

Per ottenere la numerazione delle pagine progressiva per tutta la partitura basta introdurre nel primo dei file (`son1.pmx`) il numero di pagina secondo la codifica prevista da **PMX** (*cf*r il paragrafo 2.3.17, pagina 58): la numerazione proseguirà per tutta la partitura.

4.1.2 Pentagrammi di diversa grandezza

Nelle edizioni moderne di opere che prevedono il basso continuo è consuetudine indicare la realizzazione del basso, fatta dal revisore, su un pentagramma in corpo minore. Ciò può essere ottenuto anche in **MusiX $\text{T}_{\text{E}}\text{X}$** grazie a due estensioni create da Mthimkhulu Molekwa (e-mail: *vedi* indirizzo n. 6 di figura 1.2, pag. 25) per questo scopo, recentemente entrate a far parte della distribuzione ufficiale di **MusiX $\text{T}_{\text{E}}\text{X}$** . La prima, `musixbar.tex`, permette di definire con semplicità quali pentagrammi vanno uniti mediante le linee di battuta (nel caso del basso continuo, per esempio, i due più bassi). La seconda, `curly.tex` permette di introdurre le parentesi graffe a sinistra (nel caso del basso continuo, la graffa per unire il rigo del basso con quello contenente la realizzazione del continuo in corpo minore).

Ecco come si usano. Poniamo il caso di dovere stampare una sonata per violino e continuo con **PMX**. Vi saranno tre pentagrammi: quello del basso in fondo al sistema, quello del continuo, in corpo minore, immediatamente sopra, unito al precedente mediante la graffa e quello del violino in alto. Per far ciò dobbiamo fare le seguenti operazioni:

- Indicare nel preambolo tre strumenti diversi (*e non due* di cui uno con doppio pentagramma)
- Invocare, all'inizio del file, le due estensioni mediante il comando $\text{T}_{\text{E}}\text{X}$ in linea del tipo 4, `\input musixbar\input curly`. Un esempio:

```
---
\let\:=\relax\input musixtex\:\sepbarrules\input pmx
\input musixbar\input curly
---
```

L'opzione `\sepbarrule\` fa in modo che le linee di battuta siano regolate individualmente.

- Introdurre i seguenti comandi immediatamente prima della battuta iniziale (in genere dopo il comando **A**):

```
\\indivbarrules\sepbarrule3\
\setsize2\smallvalue\curlybrackets{{1}{2}}\
\let\interstaffsav\interstaff\def\interstaff#1{\interstaffsav{9}\
```

Il valore `\interstaffsav{9}` regola l'estensione della graffa e può essere modificato. Il comando `\\indivbarrules\sepbarrule3` indica che il terzo pentagramma (quello del violino) deve avere linee di battuta proprie, non unite a quelle degli altri pentagrammi. La seconda linea del precedente codice contiene due comandi: il primo (`\setsize2\smallvalue`) pone in corpo minore il secondo pentagramma, il secondo (`\curlybrackets{{1}{2}}`) indica quali pentagrammi devono essere uniti dalla parentesi graffa.

Per il resto si procede nella stesura del codice come al solito. Se si usa **M-Tx** i comandi sono gli stessi, ma, come si è visto, vanno inclusi come comandi **PMX** in linea (%). Un esempio si trova nel codice riprodotto a pagina 140 (paragrafo 5.3).

4.1.3 Definizioni e ridefinizioni

Abbiamo spesso parlato di ‘macro’, ma non ne abbiamo spiegato il significato. In informatica una ‘macro’ è costituita da una sequenza di comandi rappresentati da un nome. Quando in un codice compare il nome, viene automaticamente eseguita tutta la sequenza di comandi. In \TeX le macro vengono chiamate ‘definizioni’¹ ed hanno la seguente forma:

```
\def\nuovonome{comandi da eseguire}
```

Ne abbiamo già fatto uso nelle pagine precedenti. Un esempio è fornito nel codice **PMX** di pagina 133:

```
\def\comp{\rightline{\medtype Guillaume Dufay}}
```

In questo caso il comando `\comp` stampa il nome con i caratteri voluti e nella posizione desiderata. Attenzione! Bisogna assicurarsi che i nomi scelti per le nuove macro non esistano già, perché ciò creerebbe conflitti.

È anche possibile ridefinire comandi già esistenti, attribuendo loro un nome alternativo, con `\let\nuovonome\nomecomando`. Lo si può fare per brevità. Ad esempio in **Musixlyr** il melisma è indicato con `\beginmel ... \endmel`. Per comodità possiamo ridefinire così i comandi

```
\let\BM\beginmel
\let\EM\endmel
```

e utilizzare `\BM` ed `\EM` al posto dei comandi originali. `\let` può essere anche utilizzato (come ci è già capitato di vedere nel codice di pagina 113 e come vedremo meglio fra poco) per conservare il contenuto originale di un comando che si intende modificare. Per maggiori dettagli su queste tecniche si vedano [4] (o la traduzione

¹Ci è anche capitato di dire che anche **MusiX \TeX** è una macro ed in effetti lo è nel senso che, facendo uso del linguaggio \TeX , mette a disposizione una serie di definizioni che rendono possibile la stampa musicale.

italiana [5]), [17] e soprattutto [11]. Vediamo invece per quale impiego pratico queste due tecniche possono essere utilizzate.

Qualche volta, per stampare musica antica, ci si trova di fronte ad una notazione non convenzionale, come quella già vista in figura 3.8, o come quella dell'esempio seguente (figura 4.1a) che ricavo (con modifiche) da una *Giga* di Domenico Zipoli.



FIGURA 4.1 – Ridefinizione di comandi T_EX

Si tratta di un brano scritto con due metri diversi (12/8 per la mano destra e C per la sinistra). **PMX** non mette a disposizione comandi in grado di produrre una simile stampa, perciò dobbiamo ricorrere a T_EX, per definire nuove macro o ridefinire quelle già esistenti.

In particolare abbiamo bisogno di

1. stampare indicazioni metriche diverse per ciascun pentagramma
2. fare in modo che la musica segua il doppio metro senza che **PMX** rilevi errori ritmici

Metro In **MusiX_TE_X** il metro viene stampato, se espresso da una frazione, con `\generalmeter{\meterfrac{n}{d}}` (n = nominatore, d = denominatore).

Nel nostro caso però se usiamo `\generalmeter{\meterfrac{12}{8}}` otterremmo il metro di 12/8 su ambedue i pentagrammi. Per ottenere ciò che ci serve dobbiamo utilizzare il comando

```
\setmeter1{\meterC{\meterfrac{12}{8}}}
```

`\setmeter1` indica il numero di strumento (noi abbiamo un solo strumento con due pentagrammi), `\meterC` il metro del pentagramma inferiore e `\meterfrac{12}{8}` il metro di quello superiore.

Se abbiamo bisogno di ritornare al normale metro di 12/8 su entrambi i pentagrammi (come in figura 4.1b) basta usare, all'inizio del blocco, il comando di cambio di metro previsto da **PMX**: `m128128`.

Durate Per evitare errori ritmici noi inganneremo **PMX** scrivendo la musica interamente in 12/8, poi modificheremo l'aspetto delle note della parte inferiore, eliminando i punti di valore, affinché corrisponda al metro di 4/4. Nell'ultima battuta riporteremo tutto alla normalità.

Cominciamo dalla pausa di semiminima. In **MusiX_{TEX}** è codificata così: **qp** (senza punto) **qpp** (con punto). Un problema è dato dal fatto che la pausa di semiminima capita in tutti e due i pentagrammi, quindi dobbiamo fare in modo che le modifiche siano applicate solo in alcuni casi e non sempre. Ricavo quasi testualmente da un messaggio di Don Simons alla lista di discussione [1] la tecnica da usare in simili casi:

1. conservare il significato originale della macro che si intende modificare utilizzando il comando `\let`
2. ridefinire la macro originale in modo da ottenere l'effetto desiderato
3. ripristinare la macro originale

Così la modifica avviene e immediatamente tutto torna alla normalità, con un solo comando. Nel nostro caso possiamo procedere così:

```
\def\pausanopunto{\let\qppsav\qpp\let\qpp\qp\let\qppsav\qpp}
```

Il comando `\pausanopunto` compie le seguenti operazioni:

1. salva il significato originale di `\qpp` in `\qppsav`
2. ridefinisce il comando `\qp` come `\qpp` in modo che quando **PMX** scrive nel codice **MusiX_{TEX}** in uscita `\qpp`, quest'ultimo intenda `\qp` e stampi la pausa senza punto
3. riporta tutto alla normalità (`\let\qppsav\qpp`)

Per ottenere l'effetto basterà introdurre il nostro nuovo comando davanti alla pausa che vogliamo modificare:

```
\pausanopunto\ rd4 /
```

Per quanto riguarda le note siamo più fortunati: sono da modificare solo semiminime e minime; possiamo dunque modificare l'aspetto di queste figure e ripristinare le normali condizioni quando il metro ritorna unico. Siccome però compare una semiminima anche alla mano destra (batt. 2) che deve rimanere puntata, dovremo in quel caso ritornare temporaneamente alla notazione normale.

In **MusiX_{TEX}** le semiminime vanno codificate come `\qu` (gamba su), `\ql` (gamba giù) `\zu` (solo la testa di nota). Le stesse figure con il punto di valore divengono `\qup`, `\qlp` e `\zup`. Siccome non capitano semiminime da inserire come sole teste di nota in accordi, le escluderemo dalla ridefinizione. In modo analogo la codifica delle minime rispettivamente senza e con il punto di valore va fatta così: `\hu`, `\hl`, `\zh` e `\hup`, `\hlp`, `\zhp`.

La soluzione non è farina del mio sacco (come buona parte delle strategie contenute in questo capitolo) ma viene dall'aiuto della lista di discussione [1]. Per prima cosa conserviamo i comandi originali per ripristinarli quando occorre, poi ridefiniamo le note puntate come note senza punto in modo che quando **PMX** scrive — ad esempio — `\qup`, **MusiX_{TEX}** legga `\qu` e stampi la nota senza punto. Chiamiamo `\quattro` il nuovo comando:

```

\def\quattro{%
\let\hupsav\hup\let\hup\hu
\let\hlpsav\hlp\let\hlp\hl
\let\zhpsav\zhp\let\zhp\zh
\let\qupsav\qup\let\qup\qu
\let\qlpsav\qlp\let\qlp\ql}

```

Potremmo aggiungere a mano nel codice i punti necessari (procedura non utilizzabile per le pause) con `\pt{p}` da porre appena prima della nota da modificare, dove `p` indica l'altezza della nota da introdurre a mano. Una soluzione più elegante si ottiene mediante una procedura simile a quella appena vista. Definiamo due nuovi comandi che riportano transitoriamente la notazione della semiminima alla normalità: `\quponce` (se il gambo è su) e `\qlponce` (se è giù).

```

\def\quponce{\def\qup##1{\qupsav{##1}\let\qup\qu}}
\def\qlponce{\def\qlp##1{\qlpsav{##1}\let\qlp\ql}}

```

`##1` indica una variabile, cioè l'altezza della nota. Inserendo uno dei due comandi davanti ad una nota si stampa quella nota come va stampata regolarmente e immediatamente dopo si ridefinisce il valore alterato già visto perché le note successive siano modificate.

Per riportare tutto in ordine useremo una serie di ridefinizioni opposte a quelle usate prima che chiameremo `\dodici`:

```

\def\dodici{%
\let\hup\hupsav
\let\hlp\hlpsav
\let\zhp\zhpsav
\let\qup\qupsav
\let\qlp\qlpsav}

```

Ed ecco infine il codice da cui è stata ricavata la figura 4.1.

```

---
\def\pauzanopunto{\let\qppsav\qpp\let\qpp\qp\let\qppsav\qpp}
\def\quattro{%
\let\hupsav\hup\let\hup\hu
\let\hlpsav\hlp\let\hlp\hl
\let\zhpsav\zhp\let\zhp\zh
\let\qupsav\qup\let\qup\qu
\let\qlpsav\qlp\let\qlp\ql}
\def\quponce{\def\qup##1{\qupsav{##1}\let\qup\qu}}
\def\qlponce{\def\qlp##1{\qlpsav{##1}\let\qlp\ql}}
\def\dodici{%
\let\hup\hupsav
\let\hlp\hlpsav

```

```

\let\zhp\zhpsav
\let\qup\qupsav
\let\qlp\qlpsav}
---
2 1 12 8 0 0 3 -1
1 1 20 0

bt
./
w150m
Ab

% 0
\\setmeter1{\meterC{\meterfrac{12}{8}}}\
\\quattro\
\pausanopunto\ rd4 /
r8 D"a"+16-6 [ d+ c ] /

% 1
m12800 gd2 cd4- edf //
dd4 bd cd gd /
b8 a g d+ c b ef d c b a g /

% 2
dd2 gd Rd //
ad2 bd2 zd /
fs8 g a d- g f \quponce\ gd4 rd4 /

% 2 bis
\\dodici\
m128128 dd2 gd Rd //
ad2 bd2 zd /
fs8 D"b"+16-5 g a d- g f gd4 r4d /

```

4.1.4 Giustificazione verticale della pagina

In generale i sistemi di giustificazione verticale della pagina previsti da **PMX** e da **M-Tx** danno risultati eccellenti e non richiedono aggiustamenti. **PMX** in base ai valori inseriti calcola gli opportuni comandi \TeX da introdurre nel codice in uscita all'inizio del brano e per ogni nuova pagina. **M-Tx** non fa calcoli, ma inserisce il valore fornito con gli opportuni comandi \TeX in linea. Per avere un controllo completo della giustificazione verticale bisogna però ricorrere ai comandi

di **MusiX_TE_X**. Vale la pena di riepilogare i comandi già visti² e di fornire una spiegazione di alcuni comandi **MusiX_TE_X** che ancora non conosciamo.³

PMX

AI regola lo spazio fra i pentagrammi (mediante il valore di `\interstaff`) di tutta la partitura

Ai regola lo spazio fra i pentagrammi della prima pagina

Ae pone i sistemi alla stessa distanza l'uno dall'altro senza considerare il contenuto musicale

M-Tx

Space: `i1 i2 in ...` aggiunge i Interline di spazio addizionale fra gli strumenti⁴.

MusiX_TE_X

`\Interligne` distanza verticale fra le parti inferiori di due linee successive del corrente strumento (insomma fra due righe contigue dello stesso pentagramma). Si tratta dell'unità di misura impiegata da **MusiX_TE_X** per esprimere le distanze verticali.

`\interstaff` definisce lo spazio fra i pentagrammi, cioè lo spazio fra la linea inferiore di un pentagramma e la linea inferiore del pentagramma successivo

`\setinterinstrument` definisce lo spazio addizionale fra strumenti, cioè la distanza fra la linea inferiore del precedente strumento e la linea inferiore del pentagramma superiore⁵ del corrente strumento

`\staffbotmarg` definisce lo spazio sotto il sistema, cioè sotto il pentagramma inferiore dello strumento inferiore del sistema

`\stafftopmarg` definisce lo spazio sopra il sistema, cioè sopra il pentagramma superiore dello strumento superiore del sistema

`\raggedbottom` incolla alla parte alta della pagina i sistemi rimuovendo gli spazi verticali fra i sistemi

`\normalbottom` ritorna al modo normale: i sistemi sono distribuiti uniformemente nella pagina

Va tenuto presente che questi parametri sono predefiniti o sono inseriti da **PMX** all'inizio della partitura o all'inizio di ciascun movimento. Se si intende modificarne il valore nel corso del brano allora lo si deve fare mediante il comando

```
\def\atnextline{sequenza di controllo}
```

²Cfr i paragrafi 2.3.18 e 3.1.

³Cfr [3, par. 2.20] per maggiori dettagli.

⁴Nel manuale di **M-Tx** [14] si usa, in verità, l'espressione corrispondente inglese 'interline'.

⁵Va ricordato che uno strumento, come il piano, può avere più di un pentagramma.

perché esso abbia effetto nel sistema musicale successivo a quello in cui è inserito.

Per esempio se si vuole portare a 9.5 valore di `\interstaff` a partire, mettiamo, dal terzo sistema della partitura, dobbiamo porre all'inizio di una delle battute che si trovano nel secondo sistema della partitura il comando

```
\def\atnextline{\interstaff{9.5}}
```

Se stiamo usando **M-Tx** in genere non abbiamo bisogno di preoccuparci di introdurre i comandi **MusiX_{TEX}** perché `\Space`: lo fa per noi. Se il comando contiene tanti numeri quanti sono i pentagrammi di ciascun sistema l'ultima cifra aggiunge uno spazio sotto il sistema⁶ attraverso `\staffbotmarg` mentre le altre cifre inseriscono degli spazi fra i pentagrammi del sistema attraverso `\setinterinstrument`. Se ad esempio abbiamo 3 pentagrammi, il comando `Space: 3 0 4` aggiunge `3/Interligne` sotto il primo pentagramma, `0/Interligne` sotto il secondo e `4` sotto il sistema. Simili comandi possono essere introdotti anche dopo il primo sistema per regolare la distanza fra i pentagrammi dei sistemi successivi.

Questo metodo è efficace e facile da usare, ma con una limitazione: non è utilizzabile se vogliamo modificare la distanza di pentagrammi che fanno parte dello stesso strumento (ad esempio quelli del piano), perché non definisce il valore di `\interstaff`.

Anche se stiamo lavorando con **PMX** quasi sempre possiamo ottenere ciò che vogliamo con i comandi messi a disposizione dal programma. Se però abbiamo bisogno di utilizzare i comandi in linea di **MusiX_{TEX}** (può capitare soprattutto quando abbiamo a che fare con partiture particolarmente fitte), allora la cosa è un po' più complicata, perché, come abbiamo già detto, **PMX** ricalcola ad ogni pagina la giustificazione verticale. In questo caso possiamo usare uno stratagemma suggerito tempo fa da Werner Icking. Si tratta di compiere 3 operazioni.

1. Neutralizzare il valore di `\interstaff` stabilito da **PMX**: `\interstaff` viene ridefinito e al suo posto si usa il valore di `\interstaffsav`.

```
\\let\interstaffsav\interstaff\def\interstaff#1{\interstaffsav{11}}
```

2. Definire una macro per l'inserimento *nella linea successiva* del nuovo valore di `\interstaff`.

```
\\def\Ninterstaff#1{\gdef\atnextline{\interstaffsav{#1}}}
```

3. Inserire dove necessario il valore desiderato di `\interstaff` (anche decimale).

```
\Ninterstaff{11.5}
```

⁶ Cfr [14, pag. 10] per maggiori dettagli.

Ovviamente bisogna evitare di introdurre nella partitura i comandi `AI`, `Ai` e `Ae`.

Con una procedura simile è possibile modificare anche i valori di tutti gli altri parametri di **MusiXTeX** visti sopra.

Nel paragrafo 5.4 è possibile vedere un esempio dell'uso di questi comandi e dell'effetto che essi producono (pagg. 162 e seguenti).

4.1.5 Incompatibilità fra `M-Tx` e `scor2prt`

Quando si estraggono le parti da file prodotti da `MTX` contenenti il comando `@` per la regolazione del testo, `scor2prt` produce errori⁷. Per evitarli bisogna introdurre il comando nella forma prevista da **Musixlyr**.

La seguente battuta (in cui figurano tre strumenti, uno per il canto, uno, con due voci, per la realizzazione del continuo e uno per il basso) contiene un comando per l'inserimento del testo `L:{A1}` e uno per regolare l'altezza delle liriche `@3`.

```
@3 b8 [ b c1 d ]
L:{A1}
b4 zd- zb r8+0
rbd4
b4 r8
```

Per evitare che `scor2prt` produca parti errate, essa va scritta così:

```
% Riassegna le parole in modo corretto nella parte del canto
%%3\\mtxAssignLyrics{1}{A1}\
% Sistema l'altezza delle liriche senza usare @ nella parte del canto
%%3\\mtxLyricsAdjust{1}{-6}\
% Sistema l'altezza delle liriche senza usare @ nella partitura
%%%
%%\\mtxLyricsAdjust{3}{3}\
b8 [ b c1 d ]
L:{A1}
b4 zd- zb r8+0
rbd4
b4 r8
```

Un altro errore capita con la definizione dei margini. Se si introduce il comando

```
% w176m \\hoffset -1in\advance\hoffset 17mm\
% h263m \\voffset -1in\advance\voffset 17mm\
```

che serve a mantenere un margine di 17 millimetri ad ogni bordo della pagina, nelle parti staccate viene trascritto solo

⁷ Vedi paragrafo 2.5.

w176m
h263m

Per evitare l'errore bisogna introdurre il comando nella seguente forma:

```
%% w176m
%%\hoffset -1in\advance\hoffset 17mm\
%% h263m
%%\voffset -1in\advance\voffset 17mm\
```

Si possono vedere altri esempi di simili procedure nel paragrafo 5.3.

4.2 Trucchi per PMX

PMX ha delle limitazioni. Molte di esse possono essere superate anche solo con le risorse messe a disposizione dal programma stesso. Eccone alcuni esempi.

4.2.1 Fusa (1/128)



FIGURA 4.2 – Inserimento di fuse in PMX

In PMX la durata massima consentita è la semibiscroma (1/64). Tuttavia è abbastanza semplice ottenere il valore di una fusa (1/128): basta codificarla come gruppo sovrabbondante. Ad esempio, una scala di 8 fuse dovrà essere codificata così: c1x8n d e f g a b c .

L'esempio riprodotto in figura 4.2, tratto dalle *Variazioni K 354* per piano di Mozart (var. XI, b. 17), è stato codificato così:

```
2 1 4 4 0 0 0 -3
1 1 20 0
```

bt

./

Abp

w140m

```
c4 zf za t c8- zg+ t r an2- zgf ze zc Rb /
\turn{10}\ d8+ s f1.e s e8 e- t [-1 e6 t su+10 f e d ]
[-2 e8x6n-3f en f fs g a ] [ an8x6n+1 b bn c cs d ]
[ ef3 en fn gf f6 af g f6x2n ef s+0+0+4 ] /
```

4.2.2 Tre voci su un solo pentagramma

PMX consente come si è visto l'inserimento di un massimo di due voci per pentagramma, ma, come nel caso appena visto, è possibile superare questa limitazione in modo abbastanza semplice. La soluzione in questo caso è stata proposta da Oliver Vogel e consiste semplicemente nello scrivere la terza voce su un pentagramma a parte, al di sopra di quello contenente le altre due voci, poi abbassare il pentagramma in maniera che coincida esattamente con quello sottostante mediante il comando

```
\setinterinstrument{n}{-12\Interligne}\interstaff{12}\
```

I comandi `\interstaff` e `\setinterinstrument` sono comandi già visti nel paragrafo 4.1.4. La variabile n rappresenta il numero dello strumento al di sopra del quale si trova quello virtuale. Il comando stabilisce un valore di `\interstaff` pari a 12Interligne e poi aggiunge uno spazio pari a -12Interligne (cioè sottrae tale spazio), al di sopra dello strumento a cui si riferisce il comando `\setinterinstrument`.

Ovviamente nel preambolo la voce aggiunta deve essere considerata ed andranno inoltre eseguite manualmente le opportune correzioni alle direzioni delle aste e alla spaziatura orizzontale delle note. Un esempio tratto dalla *Sarabanda* della *Suite inglese n. 2* di Bach chiarirà meglio le cose (fig. 4.3). Eccone il codice.

```
3 2 3 4 3 4 0 0
1 1 20 0
```

```
btt
./
B
Ab
w90m
```

```
\setinterinstrument{1}{-12\Interligne}\interstaff{12}\
% 1
a4 d- e /
e4 e8 d c b //
XS+.5 a4+ XS+.85 f XS+.5 e8 XS+.5 d /
cu4+ c8 b a gs /

% 2
f4 e8 d c b Rb /
a4 a2 //
XS+.5 c4 XS+.5 c2 /
a4 a2 /
```

FIGURA 4.3 – Pentagramma con tre voci in **PMX**

Come si vede nel preambolo sono previsti 3 pentagrammi e due strumenti anziché il normale rigo di piano (2 pentagrammi ed uno strumento). Nella batt. 1 è stato necessario dirigere forzatamente verso l'alto l'asta della prima nota della voce superiore (cu4+). Nella seconda voce del pentagramma superiore sono stati necessari diversi aggiustamenti per evitare collisioni con le altre note. La parte più importante del codice è però

```
\setinterinstrument{1}{-12\Interligne}\interstaff{12}\
```

che pone sopra lo strumento n. 1 uno spazio negativo di 12\Interligne (cioè sottrae tale spazio) e al tempo stesso stabilisce che la distanza fra i pentagrammi \interstaff è della medesima entità. In questo modo il terzo pentagramma si sovrapporrà esattamente al secondo, inserendo la terza linea musicale sul secondo pentagramma.

Va però usata una precauzione: **PMX** ridefinisce all'inizio di ogni nuova pagina i valori di \interstaff e di \interinstrument, i quali dunque potrebbero variare determinando una posizione errata del pentagramma virtuale nelle pagine successive alla prima. Per impedire che ciò avvenga bisogna utilizzare il sistema di giustificazione verticale suggerito nel paragrafo 4.1.4 ed evitare i comandi **PMX Ai**, **AI** ed **Ae**.

4.3 Uso di ABCPP

ABCP⁸ è un preprocessore creato da Guido Gonzato. È nato per essere utilizzato con il sistema di notazione musicale ABC e i suoi programmi, ma funziona perfettamente anche con **M-Tx** e **PMX**. In particolare esso risulta utile per due sue caratteristiche: la possibilità di stabilire delle macro per definire o ridefinire comandi e la compilazione condizionata. Una volta installato⁹ il programma funziona a riga di comando. Legge un semplice file di testo e lo restituisce in uscita con le modifiche inserite nel medesimo file di ingresso. Lo schema d'uso è

```
abcpp [opzioni] [-definizione1 -definizione2 ...] inputfile outputfile
```

⁸Per la documentazione si veda [9].

⁹Si faccia riferimento per questo alle istruzioni in italiano contenute nel programma.

Tralascio le opzioni¹⁰ perché si tratta di comandi specifici per il linguaggio ABC. Se nel file sono state incluse delle direttive condizionali, selezionando il loro nome nella riga di comando tali direttive saranno incluse nel file d'uscita, altrimenti no. Inoltre se sono presenti delle macro, esse saranno sostituite con i relativi comandi nel file d'uscita.

4.3.1 Macro

Abbiamo visto che \TeX dà la possibilità di definire o ridefinire dei comandi (paragrafo 4.1.3). **PMX** non permette una simile possibilità. **ABCPP** in tali casi ci viene in aiuto.

Per definire una macro si usa il comando `#define definizione comando`, così quando il programma incontra 'definizione' la sostituisce con 'comando' nel file di uscita. Per esempio se volessimo usare la notazione italiana in **PMX** (**M-Tx** già lo consente), dovremmo predisporre le seguenti macro:

```
#define do c
#define re d
#define mi e
#define fa f
#define sol g
#define la a
#define si b
```

I comandi delle singole note sono stati insomma ridefiniti. L'impiego di questa notazione va bene come esempio, ma, in verità, non mi pare molto produttivo, perché richiede un maggior numero di caratteri per introdurre la stessa musica, inoltre può creare conflitti con altri simboli testuali. Se ad esempio da qualche parte si trova la didascalia "violino solo" in uscita essa sarà sostituita con "violino go" perché tutti i "sol" saranno rimpiazzati da "g". Per evitarlo dobbiamo scrivere la didascalia nella forma `violino \solo`. Il simbolo `\` serve infatti per disattivare localmente una macro. Le macro possono essere però utili in presenza di comandi ripetitivi e lunghi o, come vedremo fra poco, nella trasposizione (paragrafo 4.3.3).

4.3.2 Compilazione condizionata

Questa funzione serve a fare in modo che le parti scritte nel file di entrata all'interno di una direttiva condizionale siano ricopiate nel file in uscita solo se tale direttiva è richiamata fra le definizioni al momento della compilazione, cioè se il suo simbolo è definito. Le direttive previste sono:

`#ifdef def1 (def2 ...)` le linee che seguono il comando e che precedono `#endif` saranno ricopiate nel file di uscita solo se il simbolo che le indica è definito.

¹⁰Una lista completa si può comunque avere lanciando il comando in questo modo: `abcpp -h`.

#endif termina una condizione **#ifdef** o **#ifndef**.

#ifndef def1 (def2 ...) agisce al contrario di **#ifdef**, cioè solo se il simbolo non è definito.

#else inizia una condizione alternativa a **#ifdef** o **#ifndef**

#elifdef def1 (def2 ...) è l'equivalente di **#else** seguito da **#ifdef**

#elifndef def1 (def2 ...) è l'equivalente di **#else** seguito da **#ifndef**

Se non vogliamo che un comando presente in partitura sia ricopiato nelle parti¹¹ possiamo introdurlo in modo condizionale, ad esempio:

```
#ifdef PARTITURA
\input musixbar\input curly
#endif
```

La compilazione dovrà essere effettuata in due modi diversi:

```
abcpp -PARTITURA file.pmp file.pmx
```

per ottenere il codice **PMX** della partitura, in cui sono inclusi i comandi posti in forma condizionale;

```
abcpp file.pmp file_prt.pmx
```

per ottenere quello da cui saranno ricavate le parti, da cui i suddetti comandi sono esclusi.

Questa procedura è utile ogni volta che vogliamo ottenere risultati diversi a partire dallo stesso codice sorgente. Sappiamo, ad esempio che **PMX** non tiene in considerazione gli abbellimenti nella creazione del file midi. Se al contrario vogliamo che un abbellimento sia correttamente eseguito nel file midi dobbiamo scrivere due volte la battuta, una volta in maniera che sia stampata correttamente ed un'altra perché produca un file midi corretto, in cui siano sostituite le note da eseguire all'abbellimento.



FIGURA 4.4 – J. S. Bach, *Toccata e fuga in re minore*, incipit

La figura 4.4 ci offre un esempio tratto da un incipit che tutti riconosceranno. Per produrre il file midi corretto dobbiamo usare il seguente codice **ABCP**:

¹¹Abbiamo ad ogni modo già visto che anche **PMX** ha comandi specifici per tale scopo: *cfr* paragrafo 2.5.

```

1 1 2 4 0 6 0 -1
1 1 20 0.0

t
./
Ab
It64
w70m
#ifdef MIDI
a65 g ad1 r3 g6 f e d cs3 d1 r r8 of Rz /
#else
a85 om r3 g6 f e d cs3 d1 r r8 of Rz /
#endif

```

Poi (supponiamo di chiamare il file `mord.pmp`) dobbiamo compilare con

```

abcpp mord.pmp mord.pmx
pmxab mord
...

```

per ottenere la stampa corretta (figura 4.4) e con

```

abcpp mord.pmp mord_mid.pmx
pmxab mord_mid

```

per ottenere il file midi voluto (come in figura 4.5).



FIGURA 4.5 – J. S. Bach, *Toccata e fuga in re minore*, incipit: file midi

Ulteriori esempi di compilazione condizionata con **ABCPP** si trovano nel paragrafo 5.3.

4.3.3 Trasposizione con ABCPP e PMX

Come si è visto nel paragrafo 2.3.15 **PMX** permette assai semplicemente la trasposizione, ma non permette di annotare sulla stessa partitura parti trasposte e parti in tonalità di impianto. In genere non si tratta di una grande limitazione: si stampa la partitura con tutti gli strumenti nella sola tonalità di impianto (con l'impiego della notazione relativa degli accidenti **Ar**) e si inseriscono i comandi per la trasposizione nelle sole parti degli strumenti traspositori (es: `%2 K-2+3`).

Se vogliamo che la partitura contenga le parti già trasposte è possibile seguire il sistema proposto da Cornelius Noack nel suo manuale [15, pag. 97], cioè inserire una diversa armatura di chiave per gli strumenti traspositori mediante il comando

PMX in linea `setsign{m}{+/-n}` (vedi paragrafo 4.1.3) e scrivere le parti trasposte come sono scritte e non come dovrebbero suonare. Questa procedura ha due svantaggi: i file midi prodotti saranno dissonanti; non sarà possibile trasporre in altre tonalità le parti.

ABCP ci offre una possibilità alternativa, un po' macchinosa ma efficace, per avere i vantaggi della procedura appena descritta ed evitare gli svantaggi.

Supponiamo di avere tre strumenti che devono suonare all'unisono una scala di la minore: lo strumento inferiore non è traspositore, quello intermedio è uno strumento in mi bemolle e traspone di una terza minore (legge la scala in fa diesis minore), quello superiore è uno strumento in la e traspone di una sesta minore (legge in do minore). Ecco come fare:

1. definire due diversi gruppi di nomi per i due strumenti traspositori ($XA = f$, $XB = g$ ecc., ed $YA = c$, $YB = d$ ecc.);
2. porre le macro così create all'interno di una direttiva condizionale in modo che se essa è definita il file in uscita conterrà gli strumenti trasposti come si ottengono con il metodo di Noack, altrimenti tale file non conterrà trasposizioni;

3. Stabilire una diversa armatura per gli strumenti traspositori con il comando


```
\\setsign{2}{+3}\\setsign{3}{-3}
```

e porre tale comando all'interno di una direttiva condizionale;

4. scrivere le parti che vanno trasposte facendo uso dei nomi precedentemente definiti (XA ecc. per la parte intermedia e YA ecc. per quella superiore);
5. compilare con il comando

```
abcpp -TRASP scala.pmp scala_tr.pmx
```

per ottenere (dopo gli altri consueti passi) la partitura contenente le parti trasposte (figura 4.6);

6. compilare con il comando

```
abcpp scala.pmp scala.pmx
```

per ottenere (dopo i necessari passi) la partitura nella sola tonalità d'impianto (figura 4.7) dalla quale saranno poi estratte le parti (figure 4.8, 4.9 e 4.10) e i file midi corretti.

Ecco il codice necessario per ottenere questo risultato.

```
#ifdef TRASP
#define XA f
#define XB g
#define XC a
#define XD b
#define XE c
#define XF d
```

la

mi bem.

non trasp.

FIGURA 4.6 – Trasposizione con ABCPP: partitura da stampare

la

mi bem.

non trasp.

FIGURA 4.7 – Trasposizione con ABCPP: part. per midi ed estrazione delle parti

non trasp.

FIGURA 4.8 – Trasposizione con ABCPP: parte 1

mi bem.

FIGURA 4.9 – Trasposizione con ABCPP: parte 2

la

FIGURA 4.10 – Trasposizione con ABCPP: parte 3

```
#define XG e
#define YA c
#define YB d
```

```

#define YC e
#define YD f
#define YE g
#define YF a
#define YG b
#else
#define XA a
#define XB b
#define XC c
#define XD d
#define XE e
#define XF f
#define XG g
#define YA a
#define YB b
#define YC c
#define YD d
#define YE e
#define YF f
#define YG g
#endif
3 3 4 4 0 6 0 0
1 1 20 .1
non trasp.
mi bem.
la
btt
./
Abr
B
\\sepbarrules\
\\nobarnumbers\
w120m
#ifdef TRASP
%%
\\setsign{2}{+3}\\setsign{3}{-3}\
#endif
It92ibaclob
%1It92iba
%2It92icl
%3It92iob
a4- b c d | e fs gs a | a gn fn e | d c b a /
%2 K-2+3
XA4 XB XC XD | XE XF s XG s XA | XA XG n XF n XE | XD XC XB XA /
%3 K+2-3
YA4 YB YC YD | YE YF s YG s YA | YA YG n YF n YE | YD YC YB YA /

```


Capitolo 5

Codici e musiche

Ecco alcuni esempi di partiture stampate con **PMX** o con **M-Tx**, seguite dai relativi codici. Contengono molti comandi, anche complessi, visti in questa guida e potranno dunque servire da esempio per chi si accosta a questo sistema di stampa musicale.

5.1 Dufay, *Kyrie* (PMX)

```
---
\font\tit = cmcsc10 scaled \magstep 5
\font\dat = cmr12
\def\comp{\rightline{\medtype Guillaume Dufay}}
\def\data{\rightline{\medtype (1400?--1474)}}
---
2 1 3 4 3 4 0 0
1 6 20 0

bt
./
Abep
\\def\writebarno{\ifnum\barno>1\lrlap{\oldstyle\the\barno\barnoadd}\fi}%\
\\def\shiftbarno{0\Interligne}\
Tt
{\tit Kyrie}
Tc
\ vbox{\comp\data}
It92iororb60:68
% 1-6
h-4
Kyrie
dd23 | a23 bf4 | c24 d4 | ad23 | dd24 | a23 d44 //
a23 d4 | c2 d4 | e2 d4 | csd2 | f2 f4 | e2 f4 /
a24 a4 | e2 d4 | g2 f4 | e2 r4 | ad4 g8 a b | cd45 a84 b c85 /

% 7-9
c24 bf43 | ad2 | g2 d4 //
```

```
e24 d4 | e d8 c4 bf8 | d4 r+7 d /
a84 bf a g g f | ad2 | bf2 a4 /
```

```
% 10-12
e2 d4 | g4 bf a | gd2 Rd //
g8- f g e f4 | r8+8 [ d+ d cs c bn ] | dd2 /
g2 a4 | g2 fs4 | gd2 /
```

```
% 13-18
L3Mc+4
h-5
Christe
dd23 | fd2 | gd2 | d2 d4 | e2 d4 | g2 a4 //
dd24 | c2 f8 e | d4 c bf | ad2 | g2 f4 | r8+7 [ d+ c b c d ] /
ad24 s | ad2 s | r8 [ bf b a b g ] | fsd2 | g2 a4 | d- e f /
```

```
% 19-24
bf23 a4 | gd2 | d24 c4 | bf43 a g | d f e | dd2 Rd //
g4-r d+ c | d2 r4 | f2 e4 | d c b | a2 gs4 | ad2 /
g24 f4 | g r g | a bf a | g8 f4 e8 g d | f e c d cs b | dd2 /
```

```
% 25-28
L5Mc+4
h-5
Kyrie
d23 g4 | d r8 d e f | g4 f ef | d r+0 r+0 //
d24 ef4 | d2 c4 | bf2 c4 | dd2 /
a24 g4 | f2 g4 | bf a g | fsd2 /
```

```
% 29-32
g2 a4 | g2 f4 | g2 a4 | d2- d4 //
d2 cs4 | d2 a4 | bf2 c4 | ad2 /
r8 [ g g fs f e ] | g4 f8 g a bf | g4 f e | d r r /
```

```
% 33-34
g43 f g | a f e //
bf4 a g | X.3S f a8 g4 f8 /
e84 d4 c8 bf4 | a8 c4 d8 c4 /
```

```
% 35-36
d4 d+ c | d f e //
a4 X.4S b8f<.4 g4 a8 | X.4S d4- X.4S d8+ c4 X.75 bf8 /
d8 f4 g8 e4 | f8 d4 a8+ g4 /
```

```
% 37-39
d2 c4 | dd2 | gd2- //
d2 e4 | fd2 of-2 | dd2 of-2 /
a4 b c | ad2 of | gd2 of /
```

KYRIE

Guillaume Dufay
(1400?–1474)

Kyrie

The first system of the Kyrie section, measures 1-6. The music is in 3/4 time. The right hand features a melodic line with eighth and quarter notes, while the left hand provides a harmonic accompaniment with chords and single notes.

The second system of the Kyrie section, measures 7-12. The melodic line continues with eighth notes and quarter notes, showing some chromatic movement. The left hand accompaniment remains consistent with the first system.

Christe

The third system of the Kyrie section, measures 13-18. The section is labeled 'Christe'. The melodic line begins with a half note followed by eighth notes. The left hand accompaniment continues with a steady rhythmic pattern.

The fourth system of the Kyrie section, measures 19-24. The melodic line continues with eighth and quarter notes. The left hand accompaniment features a mix of chords and single notes.

Kyrie

The fifth system of the Kyrie section, measures 25-31. The section is labeled 'Kyrie'. The melodic line continues with eighth and quarter notes. The left hand accompaniment features a mix of chords and single notes.

The sixth system of the Kyrie section, measures 32-37. The melodic line continues with eighth and quarter notes. The left hand accompaniment features a mix of chords and single notes. The system concludes with a double bar line.

5.2 Vivaldi, *Coro* (M-Tx)

```

Title: \vbox{\titA\titB}
Composer: \vbox{\comp\data}
Flats: 2
Meter: 3/8
Pages: 1
Systems: 2
Style: SATB4
Size: 16
Space: 6 6 6 12

%%\font\rxii = cmr12
%%\font\tixii = cmti12
%%\def\titA{\centerline{Mundi rector}}
%%\def\titB{\centerline{\rxii {\tixii Juditha Triumphans}, I, 27}}
%%\def\comp{\rightline{\medtype Antonio Vivaldi}}
%%\def\data{\rightline{\rxii (1678--1741)}}
%%\let\endpiecesav\endpiece\
%%\def\endpiece{\endpiecesav\input judt}\
%%\font\rix = cmr9
%%\rix

{sopA}={altoA}={tenorA}={basA}
Mun-di Rec-tor de Cae-lo mi-can-ti
Au-di pre-ces, au-di pre-ces, et su-sci-pe vo-ta
Quae de cor-de pro te di-mi-can-ti
Sunt pie-ta-tis in si-nu de-vo-ta.

{sopB}={altoB}={tenorB}={basB}
In Ju-di-ta tuae le-gi di-ca-ta
Flam-mas dul-cis, flam-mas dul-cis, tui_a-mo-ris-ac-cen-de
Fe-ri-ta-tis sic hos-tis do-ma-ta
In Be-thu-liae spem pa-cis in-ten-de.

{sopC}={altoC}={tenorC}={basC}
Re-di, re-di iam Vic-trix pu-gnan-do
In ci-li-cio, in ci-li-cio in pre-ce ri-vi-ve
De_Ho-lo-fer-ne sic ho-die trium-phan-do
Pia Ju-di-tha per sae-cu-la vi-ve.

%%Abp
%%B
%%\def\writebarno{\ifnum\barno>1\lrlap{\oldstyle\the\barno\barnoadd}\fi}%
%%\def\shiftbarno{0\Interligne}
%%It58ibaclobobb60:70:58:68T+0-12+0+0
% 13
%%h-4
%%{\ppff ~~~Allegro}
r8 d+ e | e d d | [ c1 b ] c8 a | b g4
L: {sopA,sopB,sopC}
r8 g g | g g g | fs f f | d d4

```

```

L: {altoA,altoB,altoC}
r8 b c | c b b | a a c | b b4
L: {tenorA,tenorB,tenorC}
@-1 r8 g g | g g g | a d- d | g g4
L: {basA,basB,basC}

%17-20
r8 b b | [ b1 a ] b4 | r8 c c | [ c1 b ] c8 e
r8 f f | [ g1 f ] g4 | r8 g g | [ a1 g ] a8 c
r8 d d | b b4 | r8 g g | c c e
r8 d d | e e4 | r8 en e | f f f

%21-24
d8 e c | (~ [ b1 a b c d e ] | [ f b- ] c4 oT0 )~ | bd4 :|:
b8 c a | (~ [ bd8 a1 b c ] | d b a4 oT0 )~ | bd4 :|:
f8 g f | ( fd4 | fd4 ) | fd4 :|:
b8 e- f | (~ ( bd4- | b8 ) f4+ )~ | bd4- :|:

%25-28
%%L2
r8 bn b | [ c1 bn ] c8 g | d+ d d | [ e1 d ] c4 |
r8 g g | g g g | g g g | g g4 |
r8 d d | e e e | d d d | c c4 |
r8 g+ g | c- c c | bn b b | c e4 |

%29-31
c8 b4 | b8 a c | b c a |
a8 g4 | g8 fs a | g a fs |
fs8 d4 | en8 fs f | d ef d |
d8 d4 | cs8 d d | g c- d |

%32-34
(~ [ g1 fs g a b c ] | [ d g- ] a4 oT0 )~ | gd4 of :|
(~ [ gd8 fs1 g a ] | [ b g ] fs4 oT0 )~ | gd4 of :|
( dd4 | dd ) | dd of :|
{~ ( gd4 | g8 ) d4 }~ | gd- of :|

%%%%%%%%%%%%%% Testo scritto in un file a parte
%%%%%%%%%%%%%% judt.tex
\font\big = cmcsc10 scaled \magstep 5
\leftskip = 3 cm
\font\lyr = cmti10
\lyr

\def\lyroffs{5mm}

\vskip 10 mm
\halign{\hskip 12mm # \hfill & # \hfill & # \hfill & # \hfill \cr

Mundi Rector de Caelo micanti
& In Judita tuae legi dicata
& Redi, redi iam Victrix pugnando \cr

```

```
Audi preces et suscipe vota  
& Flammas dulcis tui amoris accende  
& In cilicio in prece rivive \cr
```

```
Quae de corde pro te dimicanti  
& Feritatis sic hostis domata  
& De Holoferne sic hodie triumphando \cr
```

```
Sunt pietatis in sinu devota.  
& In Bethuliae spem pacis intende.  
& Pia Juditha per saecula vive. \cr  
}  
\end
```

Mundi rector

Juditha Triumphans, I, 27

Antonio Vivaldi
(1678–1741)

Allegro

Mundi Rector de Caelo micanti Audi preces, audi preces, et suscipe vo - - ta
In Ju-dita tuae le-gi di-cata Flammas dulcis, flammis dulcis, tui amoris accen - - de
Redi, redi iam Victrix pugnando In ci - li - cio, in ci - li - cio in prece ri - vi - - ve

Mundi Rector de Caelo micanti Audi preces, audi preces, et suscipe vo - - ta
In Ju-dita tuae le-gi di-cata Flammas dulcis, flammis dulcis, tui amoris accen - - de
Redi, redi iam Victrix pugnando In ci - li - cio, in ci - li - cio in prece ri - vi - - ve

Mundi Rector de Caelo micanti Audi preces, audi preces, et suscipe vo - - ta
In Ju-dita tuae le-gi di-cata Flammas dulcis, flammis dulcis, tui amoris accen - - de
Redi, redi iam Victrix pugnando In ci - li - cio, in ci - li - cio in prece ri - vi - - ve

Mundi Rector de Caelo micanti Audi preces, audi preces, et suscipe vo - - ta
In Ju-dita tuae le-gi di-cata Flammas dulcis, flammis dulcis, tui amoris accen - - de
Redi, redi iam Victrix pugnando In ci - li - cio, in ci - li - cio in prece ri - vi - - ve

13

Quae de cor-de pro te di-mi-can-ti Sunt pie-ta-tis in si-nu de-vo - - - ta.
Fe-ri-ta-tis sic hostis do-ma-ta In Be-thuliae spem pacis in-ten - - - de.
De Holo-fer-ne sic hodie triumphando Pia Ju-ditha per saecu-la vi - - - ve.

Quae de cor-de pro te di-mi-can-ti Sunt pie-ta-tis in si-nu de-vo - - - ta.
Fe-ri-ta-tis sic hostis do-ma-ta In Be-thuliae spem pacis in-ten - - - de.
De Holo-fer-ne sic hodie triumphando Pia Ju-ditha per saecu-la vi - - - ve.

Quae de cor-de pro te di-mi-can-ti Sunt pie-ta-tis in si-nu de-vo - - - ta.
Fe-ri-ta-tis sic hostis do-ma-ta In Be-thuliae spem pacis in-ten - - - de.
De Holo-fer-ne sic hodie triumphando Pia Ju-ditha per saecu-la vi - - - ve.

Quae de cor-de pro te di-mi-can-ti Sunt pie-ta-tis in si-nu de-vo - - - ta.
Fe-ri-ta-tis sic hostis do-ma-ta In Be-thuliae spem pacis in-ten - - - de.
De Holo-fer-ne sic hodie triumphando Pia Ju-ditha per saecu-la vi - - - ve.

Mundi Rector de Caelo micanti In Juditha tuae legi dicata Redi, redi iam Victrix pugnando
Audi preces et suscipe vota Flammas dulcis tui amoris accende In cilicio in prece rivive
Quae de corde pro te dimicanti Feritatis sic hostis domata De Holoferne sic hodie triumphando
Sunt pietatis in sinu devota. In Bethuliae spem pacis intende. Pia Juditha per saecula vive.

5.3 Caccini, *Amor l'ali m'impenna* (ABCPP)

Il codice che segue non è (come nelle precedenti edizioni di questa guida) un puro codice **M-Tx**, ma contiene direttive condizionali di **ABCPP**. Ciò complica un po' le cose, ma consente di estrarre correttamente le parti e di stampare alternativamente sia la partitura d'esecuzione con il basso continuo realizzato sia la partitura utext su due soli pentagrammi.

Le stampe che seguono sono state ricavate dal codice con la seguente procedura.¹

Partitura con realizzazione del basso continuo

```
abcpp -CNT imp.mtp imp.mtx
prepmx imp      |
pmxab imp      | compilazione
tex imp        | normale
musixflx imp   | imp.mtx --> imp.ps
tex imp        |
dvips imp      |
```

Partitura urtext

```
abcpp imp.mtp imp_ur.mtx
poi la normale procedura imp_ur.mtx --> imp_ur.ps
```

Parte del basso

```
abcpp -BASS imp.mtp imp_basso.mtx
scor2prt imp_basso
prepmx imp_basso1
poi la normale procedura imp_basso1.pmx --> imp_basso1.ps
```

Parte del soprano

```
abcpp imp.mtp imp_ur.mtx (ma il passaggio è stato già effettuato)
scor2prt imp_ur
prepmx imp_ur2
poi la normale procedura imp_ur2.pmx --> imp_ur2.ps
```

Parte del cembalo

¹Se si vuole usare il pacchetto di legature postscript tipo M si deve aggiungere alla riga di comando per **ABCPP** **-MUSIXPSS**. Per esempio per la partitura con basso continuo il comando diverrà `abcpp -CNT -MUSIXPSS imp.mtp imp.mtx`. Senza tale opzione la partitura sarà compilata con legature postscript tipo K.

```

abcpp -CEMB imp.mtp imp_cemb.mtx
scor2prt imp_cemb
prepmx imp_cemb1
poi la normale procedura imp_cemb1.pmx --> imp_cemb1.ps

```

```

Title: {\tit {Amor l'ali m'impenna}}
Composer: \vbox{\compA\compB}
Meter: C
# Preambolo per la parte del cembalo
#ifdef CEMB
Style: Singer Piano
Singer: Voices Sop; Vocal; Clefs G
Piano: Voices LH,RH1 RH2; Continuo; Clefs G F
# Preambolo per la partitura con realizzazione del continuo
#elseifdef CNT
Style: Singer Cont Bass
Singer: Voices Sop; Vocal; Clefs G
Cont: Voices RH1,RH2; Clefs G
Bass: Voices B; Clefs F
# Preambolo per l'edizione urtext
#else
Style: Singer Bass
Singer: Voices Sop; Vocal; Clefs G
Bass: Voices B; Clefs F
#endif
Flats: 1
Systems: 12
# Comandi di impaginazione condizionali per le diverse stampe
#ifdef CNT CEMB
Pages: 3
#else
Pages: 2
#endif
#ifdef CNT
Space: 4 1
#else
Space: 3
#endif
Size: 20
Indent: 0.10
#ifdef CNT
Name: {\it{Soprano}} ~ {\it{Continuo}}
#elseifdef CEMB
Name: {\it{Soprano}} {\it{Continuo}}
#else
Name: {\it{Soprano}} {\it{Basso}}
#endif

# Impega curly brackets solo nella partitura con il continuo
#ifdef CNT
%\input musixbar\input curly

```

```

#endif
%\font\data = cmr10
%\font\tit = cmb10 scaled \magstep 4
%\font\rxii = cmr12
%\font\mov = cmmib10 scaled \magstep 1
%\font\num = cmr9
%\def\compA{\rightline{\rxii Giulio Caccini}}
%\def\compB{\rightline{\data (1550--1618)}}
%\def\writebarno{\ifnum\barno>1\lrlap{\oldstyle\the\barno\barnoadd}\fi}%
%\def\shiftbarno{0\Interligne}
%\def\BN#1#2{\lffset{.3}{\Figu{#1}{#2}}}\

```

{Aria}

A-mor l'a-li m'im-pen-na.
A-mor dol-ce, a-mor ca-ro, a-mor fe-li-ce.
Tal che non spe-ro pi{\u} n{\e} pi{\u} mi li-ce.
Pas-so nem-bi_e pro-cel-le,
pas-so'l ciel e le stel-le,
del pia-cer que-st'\e'l re-gno.
Ah, mia for-tu-na non se l'ab-bia a sde-gno.
Que-sto, que-sto m'ac-co-ra:
ch'al-tri ca-deo,
ch'al-tri ca-deo dal pa-ra-di-so_an-co-ra.
Ah, mia for-tu-na non se l'ab-bia a sde-gno.
Que-sto, que-sto m'ac-co-ra:
ch'al-tri ca-deo,
ch'al-tri ca-deo dal pa-ra-di-so_an-co-ra.

```

# Se si compila per l'uso delle legature PS tipo M
#ifdef MUSIXPSS
%\input musixpss\relax\
%\psslurmode{auto-pdfTeX-dvips}%\
#endif
%\w176m
%\hoffset -1in\advance\hoffset 17mm\
%\h263m
%\voffset -1in\advance\voffset 17mm\
%\Ab
# Se si compila per l'uso delle legature PS tipo K
#ifndef MUSIXPSS
%\Ap
#endif
%\B
# Comandi midi condizionali
#ifdef CNT
%\t86ivchaobb58:70:64
#elifdef CEMB
%\t86ihahaobb58:70:64
%%1 S6P1
%%1 AI.9
#else

```

```

%%It86ivcobb58:64
#endif
# Solo nella partitura con il continuo...
#ifdef CNT
# Usa curly e stampa il continuo in corpo minore
%% \setsize2\smallvalue\curlybrackets{{1}{2}}\
# Stabilisce un nuovo valore di interstaff
%%\let\interstaffsav\interstaff\def\interstaff#1{\interstaffsav{9}\
# Traccia le linee di battuta separate per ogni strumento
%%\indivbarrules\sepbarrule3\
#else
# Unisce i pentagrammi con la parentesi quadra nell'urtext
%%%%
%% \songtop{2}\songbottom{1}\
#endif
% Bar 1
# Usa i comandi Musixlyr per assegnare il testo
# e per regolarne l'altezza nella parte del soprano...
%%2 \mtxAssignLyrics{1}{Aria}\
%%2 \mtxLyricsAdjust{1}{-6}\
# nella partitura con il continuo...
#ifdef CNT
%%%%
%% \mtxLyricsAdjust{3}{2}\
#else
# e in quella urtext
%%%%
%% \mtxLyricsAdjust{2}{1}\
#endif
# Ignora la cifratura del basso nelle parti e nella partitura
#ifdef CNT BASS CEMB
%%F
#endif
f0
L: {Aria}
# Realizzazione del basso continuo: in ogni battuta
#ifdef CNT CEMB
f0
c0 za
#endif
f0

% Bar 2
f2 ( f2
#ifdef CNT CEMB
f2 f4 e
a2 zd c zf-
#endif
d2 - a2 6

% Bar 3
f4 ) f8 g8 (~ [ a1 b1 c1 b1 ] [ a8 g8 ]

```

```

#ifdef CNT CEMB
d4.e f.e
b2 zf f zc+
#endif
b2 a2 6

% Bar 4
[ fd8 g1 ] [ f8 g8 ] [ a1 ( g g8 ) ] [ a1 ( f f8 ) ] )~
#ifdef CNT CEMB
d0e
cd2 zf- g4 zbn
#endif
d2 7 ( d4 7 d4 #6 )

% Bar 5
g2 r4 g8 g8
#ifdef CNT CEMB
e2 g
c2 zg e+ zc
#endif
c2 c2

% Bar 6
b2 g4 d8 d8
#ifdef CNT CEMB
b4 zd- a+ zc- g2+
g2e+ d zb
#endif
g2 - g2

% Bar 7
f4 [ f1 e1 f1 g1 ] a4 [ e1 d1 e1 f1 ]
#ifdef CNT CEMB
fd2 za- g4 ze+
de2 c
#endif
d2+ - a4 c4

% Bar 8
g0
#ifdef CNT CEMB
d0 zg-
ce2 bn
#endif
# Stampa la cifratura del basso a due cifre solo in partitura
#ifdef CNT CEMB BASS
g2 t g t
#else
\BN{1}{11}\ g2 t \BN{1}{10}\ g t
#endif

% Bar 9

```

```

g0
#ifdef CNT CEMB
e0+
c0 zg
#endif
c0

% Bar 10
c4 g8 g8 b4 [ a1 b1 c1 b1 ]
#ifdef CNT CEMB
g2 g
c2 ze d zb
#endif
c2 g2 -

% Bar 11
ad4 d8- f4 [ e1 f1 g1 f1 ]
#ifdef CNT CEMB
f2.g
a2 zd d zb
#endif
d2+ - b4 g4 -

% Bar 12
e0
#ifdef CNT CEMB
\zw e\ dr2 cs
a2d-2 g4
#endif
#ifdef CNT CEMB BASS
(5t a2 (6t a4 )5t a 7 )6t
#else
\BN{0}{11}\ (5t a2 \BN{-1}{\sharpfig}\BN{0}{~10}\ (6t a4 )5t a 7 )6t
#endif

% Bar 13
d2 r4 fs8 f8
#ifdef CNT CEMB
d2 fs
fs2 zd+ d za
#endif
d2 # d2

% Bar 14
g4 d8 d8 (~ [ e8 f1 e1 ] [ f1 e1 d1 e1 ] )~
#ifdef CNT CEMB
g4 fn e2
b2 zd c zg
#endif
g2- c2

% Bar 15

```

```
f2 r4 fs8 f8
#ifdef CNT CEMB
f2 fs
a2 zc d za
#endif
f2 d2

% Bar 16
g4 d8 d8 [ e8 f8 fd8 e1 ]
#ifdef CNT CEMB
g4 fn e2
b2 zd c zg
#endif
g2- c2

% Bar 17
f4 f8 g8 a2
#ifdef CNT CEMB
f2 f
a2 zc c za
#endif
f2 f4 e8 d8

% Bar 18
cd8 c1 (~ [ b1 a1 g1 f1 ] g2 )~
#ifdef CNT CEMB
e4 ( f f ) e
g2 zc c zg
#endif
#ifdef CNT CEMB BASS
c4 t c4 t c4 c4
#else
c4 t \BN{0}{11}\ c4 t \BN{0}{11}\ c4 \BN{0}{10}\ c4
#endif

% Bar 19
f0
#ifdef CNT CEMB
f0
a0 zc
#endif
f0-

% Bar 20
ad2 g8 f8
#ifdef CNT CEMB
f4 ( c c ) bn
a4 g ( f f )
#endif
f4+ e4 ( d4 d )

% Bar 21
```

```

e8 d8 e8 f8 g2
#ifdef CNT CEMB
c2 r8+0 e d c
g2 ze rb
#endif
( c2 c8 ) c8 b8 a8

% Bar 22
g4 d4 d2
#ifdef CNT CEMB
bnr4 ( c c ) b
g0 zd+
#endif
#ifdef CNT CEMB BASS
g4 t g t g4 t g t
#else
%%%% Le due righe seguenti vanno riunite in una sola riga
\BN{1}{\sharpfig}\BN{2}{~10}\ g4 t \BN{2}{11}\ g t \BN{2}{11}\
  g4 t \BN{1}{\sharpfig}\BN{2}{~10}\ g t
#endif

% Bar 23
c0
#ifdef CNT CEMB
e0
c0 zg
#endif
c0

% Bar 24
gd4+ e8 a4 g8 f8
#ifdef CNT CEMB
e2 e4 f
g2 zc c za
#endif
c2 a4 f4

% Bar 25
g2 f8 f8 f8 e8
#ifdef CNT CEMB
e2 f
g2 zc c za
#endif
c2+ f2-

% Bar 26
d2 r8 b8+ b a
#ifdef CNT CEMB
f2 g
b2 zd d zb
#endif
b4 b8 a8 g2 -

```

```

% Bar 27
g8 e8 e8 f8 g4 [ f1 g1 a1 b1 ]
#ifdef CNT CEMB
g4 f e ( f
c2 ze c zg
#endif
#ifdef CNT CEMB BASS
c2 c4 c4
#else
c2 c4 \BN{0}{11}\ c4
#endif

% Bar 28
g0
#ifdef CNT CEMB
f2 ) cr4 b
g2 zc e zg-
#endif
#ifdef CNT CEMB BASS
(5t c2 (6t c4 )5t c 7 )6t
#else
\BN{0}{11}\ (5t c2 (6t \BN{0}{10}\ c4 )5t c 7 )6t
#endif

% Bar 29
f0
#ifdef CNT CEMB
f0+
a0
#endif
f0-

% Bar 30
ad2 g8 f8
#ifdef CNT CEMB
f4 ( c c ) bn
a4 g ( f f )
#endif
f4+ e4 ( d 7 d #6 )

% Bar 31
e8 d8 e8 f8 g2
#ifdef CNT CEMB
c2 r8+0 e d c
g2 ze rb
#endif
( c2 c8 ) c8 b8 a8

% Bar 32
gd8 (~ g1 [ f1 e1 d1 c1 ] )~ d2
#ifdef CNT CEMB

```

```

bnr4 ( c c ) b
g0 zd+
#endif
#ifdef CNT CEMB BASS
g4 t g t g t g t
#else
%%%% Le due righe seguenti vanno riunite in una sola riga
\BN{0}{\sharpfig}\BN{1}{~10}\ g4 t \BN{1}{11}\ g t \BN{1}{11}\
  g t \BN{0}{\sharpfig}\BN{1}{~10}\ g t
#endif

% Bar 33
c0
#ifdef CNT CEMB
e0
c0 zg
#endif
c0

% Bar 34
gd4+ e8 a4 g8 f8
#ifdef CNT CEMB
e2 e4 f
g2 zc c za
#endif
c2 a4 f4

% Bar 35
g2 f8 f8 f8 e8
#ifdef CNT CEMB
e2 f
g2 zc c za
#endif
c2+ f2-

% Bar 36
d2 r8 b8+ b a
#ifdef CNT CEMB
f2 g
b2 zd d zb
#endif
b4 b8 a g2 -

% Bar 37
g2 r8 c8 c8 b8
#ifdef CNT CEMB
g2 a4 b
c2 ze c4 zf d zg
#endif
c4 c8 b8 a4 6 g4

% Bar 38

```

```

a4 b4 ( c2
#ifdef CNT CEMB
a2 a
f2 c
#endif
f2 a4 6 b4

% Bar 39
[ c8 ) (~ b1 a1 ] [ g1 f1 e1 d1 ] [ e1 d1 c1 d1 ] [ e1 f1 g1 a1 ]
#ifdef CNT CEMB
g0
c0 ze
#endif
c0

% Bar 40
[ g1 c1- d1 e1 ] [ f1 g1 a1 b1 ] [ c1 b1 a g1 ] [ a1 g1 a1 f1 ]
#ifdef CNT CEMB
e0
c0 zg
#endif
c0

% Bar 41
g0 )~
#ifdef CNT CEMB
f2 e
\zw N\ ce2d+0+1 b4
#endif
#ifdef CNT CEMB BASS
(1t c2 (2t c4 )1t c 7 )2t
#else
\BN{0}{11}\ (1t c2 (2t \BN{0}{10}\ c4 )1t c 7 )2t
#endif

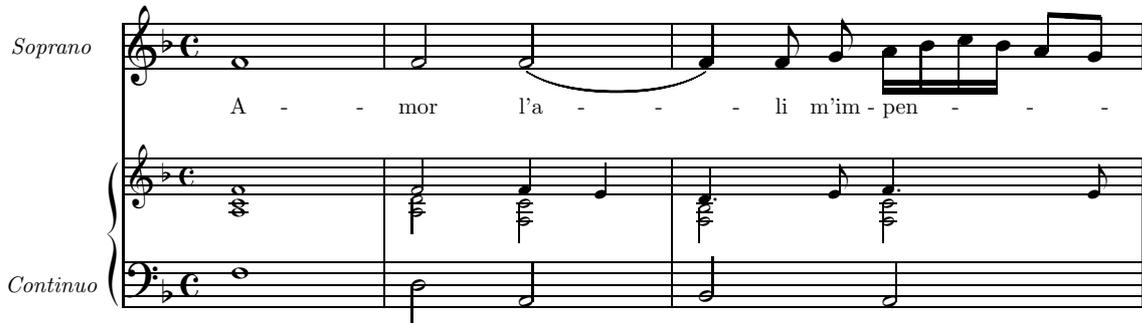
% Bar 42
f0
#ifdef CNT CEMB
f0
a0
#endif
f0-

```

Amor l'ali m'impenna

Giulio Caccini
(1550-1618)

Soprano



A - - mor l'a - - - li m'im - pen - - -

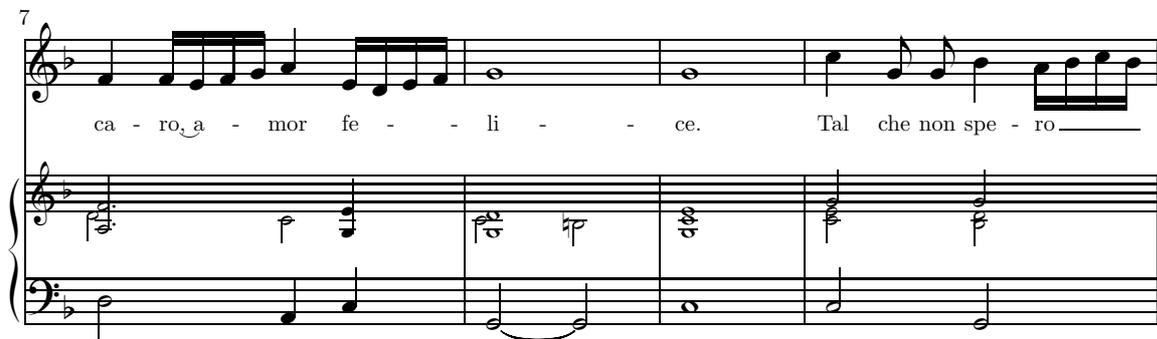
Continuo

4



- - - - - na. A - mor dol - - ce, a - mor

7



ca - ro, a - mor fe - - li - - ce. Tal che non spe - ro

11



più né più mi - - li - - ce. Pas-so nem - bi e pro-cel - -

15

le, pas - so\'l ciel e le stel - - le, del pia - cer

18

que-st\'è'l re - - - gno. Ah, mia for - tu - na non se l\'ab - -

22

bia a sde - - gno. Que - sto, que - sto m\'ac - co - ra: ch\'al - tri ca -

26

deo, ch\'altri ca - deo dal pa - ra - di - so an - co - - - ra.

30

Ah, mia for - tu - na non se l'ab - - bia a _____ sde - -

33

gno. Que - sto, que - sto m'ac - co - ra: ch'al-tri ca - deo, ch'al-tri ca -

37

deo dal pa - ra - di - so an - co - - - - -

40

- - - - - ra.

Amor l'ali m'impenna

Giulio Caccini
(1550–1618)

Soprano

A - - mor l'a - - - li m'im - pen - - -

Basso

- - - - - na. A - mor dol - ce, a - mor

ca - ro, a - mor fe - - li - - ce. Tal che non spe - ro

più né più mi - - li - - ce. Pas-so nem - bi e pro-cel - -

le, pas-so'l ciel e le stel - - le, del pia - cer

que-st'è'l re - - - gno. Ah, mia for - tu-na non se l'ab - -

4

7

11

15

18

7 7 #6 b

11 10 b

11 #10 7 #

11 11 10

22

bia a sde - - gno. Que - sto, que - sto m'ac - co - ra: ch'al-tri ca -

#10 11 11 #10

26

deo, ch'altri ca - deo dal pa - ra - di - so an - co - - - ra.

11 11 10 7

30

Ah, mia for - tu - na non se l'ab - - bia a sde - -

7 #6 #10 11 11 #10

33

gno. Que - sto, que - sto m'ac - co - ra: ch'al-tri ca - deo, ch'al-tri ca -

37

deo dal pa - ra - di - so an - co - - - - -

6 6

40

- - - - - ra.

11 10 7

Basso

Amor l'ali m'impenna

Giulio Caccini
(1550-1618)

5

9

13

17

20

23

27

30

33

36

39

Continuo

Amor l'ali m'impenna

Giulio Caccini
(1550–1618)

8

16

22

29

36

5.4 Boccherini, *Cello concerto n. 1* [estr.] (PMX)

```

6 -5 1 1 2 1 1   4 4 0 6 0 0
3 7 20 0.11
%
Vc. e Cb.
Viola
Vl.~1~e~2
Vc. solo
Corni~in~Do
battbt
./
Abp
B
% COMANDI PER L'IMPAGINAZIONE
w160m
h260m
% Raggruppa gli strumenti ad arco
\\songtop{3}\
\\songbottom{1}\
% Sistema il valore di interstaff per tutta la partitura
\\let\\interstaffsav\\interstaff\\def\\interstaff#1{\\interstaffsav{10.5}\
% Definisce il comando \\Nlinterstaff per variare il valore
% di interstaff a partire dal sistema successivo a quello
% in cui è inserito
\\def\\Nlinterstaff#1{\\gdef\\atnextline{\\interstaffsav{#1}}}\
% Imposta linee di battuta separate per ogni strumento
\\sepbarrules\
% Comandi MIDI
Iivcvavlvlvcfrv100:100:100:100:100:100g2t68
Tt
Concerto per violoncello
Tc
Luigi Boccherini
% Mov I %%%%%%%%%%%
% Batt. 1
h-3
Allegro
c83 Df c14d b3 c8 g e c e g /
c84 Df c15d b3 c8 g e c e g /
c84 Df c15d b3 c8 g e c e g /
c84 Df c15d b3 c8 g e c e g /
c83 Df c14d b3 c8 g e c e g /
c44 rb2 rb4 //
c45 r+0 r2+0 /

% Batt. 2
c83 c14d b3 c8 a f c f a /
c84 c15d b3 c8 a f c f a /
c84 c15d b3 c8 a f c f a /
c84 c15d b3 c8 a f c f a /
c83 c14d b3 c8 a f c f a /

```

```

c44 rb2 rb4 //
c45 r+0 r2+0 /

% Batt. 3
c83 c14d b3 c8 g e c r4 /
c84 c15d b3 c8 g e c r4 /
c84 c15d b3 c8 g e g c e /
c84 c15d b3 c8 g e c85 e g /
c83 c14d b3 c8 g e c r4 /
c44 rb2 rb4 //
c45 r+0 r2+0 /

% Batt. 4
r4 g43 b g /
r4 b24 b4 /
Gsf5 e8 d d4d d84 g d85 /
g85 f f4d g84 d85 f /
r4 g43 b g /
r0 /

% Batt. 5
% Modifica il valore di interstaff per la seconda pagina
\NLinterstaff{11}\
r4 c44 c c43 /
r4 g24 c4 /
Gse5 d8 c c4d g8 c e /
f85 e e4d c8 e g /
r4 c44 c c43 /
r0 /

% Batt. 6
f83 f f f g82 g g g /
c85 c c c g g g g /
a44 Gxsb a8 g1 f Gxsf e8 d1 c Gxse d8 c1 b /
Gxsg5 f8 e1 d Gxsd c8 b1 a Gxsa g8 f1 e Gxsg f8 e1 d /
f83 f f f g82 g g g /
c45 rb c g //
c45 r+0 e d /

% Batt. 7
c83 c14d b3 c8 g e c e g /
c84 c15d b3 c8 g e c e g /
c84 c15d b3 c8 g e c e g /
c84 c15d b3 c8 g e c e g /
c83 c14d b3 c8 g e c e g /
e44 rb2 rb4 //
c45 r+0 r2+0 /

% Batt. 8
c83 c14d b3 c8 a f c f a /
c84 c15d b3 c8 a f c f a /
c84 c15d b3 c8 a f c f a /

```

```

c84 c15d b3 c8 a f c f a /
c83 c14d b3 c8 a f c f a /
c44 rb2 rb4 //
c45 r+0 r2+0 /

```

```

% Batt. 9
c83 c14d b3 c8 g e c r4 /
c84 c15d b3 c8 g e c r4 /
c84 c15d b3 c8 g e g83 c e /
c84 c15d b3 c8 g e c e g /
c83 c14d b3 c8 g e c r4 /
c44 rb2 rb4 //
c45 r+0 r2+0 /

```

```

% Batt. 10
r4 g43 b g Rb /
r4 b24 b4 /
e84 d d4d g83 b d /
g84 f f4d g83 d84 f /
r4 g43 b g /
r0 /

```

```

% Mov II %%%%%%%%%%%
% Stabilisce un nuovo valore di interstaff per il secondo movimento
\\interstaffsav{9}\
% Inserisce l'a capo forzato, la voltata di pagina
% l'inizio del nuovo movimento e il nuovo numero di strumenti
L5P3Mn41234battb
% Comandi MIDI
Iivcvavlvlvcv100:100:100:100:110g2t46
% Batt. 1
h-2
Largo
m3434 f83 D"Vc. soli" f f f82 a s0 bn s0 /
rp /
rp /
f84d ot s0 e3 f s0 c8 f f f /
f83 f f f82 a s0 bn s0 /

```

```

% Batt. 2
c83 D"Vc. e Cb. uniti" c c c c c /
c84 c c c c c /
g84d ot s0 f3 g s0 c44 e8 f /
f44 e8 f g a /
c83 c c c c c /

```

```

% Batt. 3
c83 c c c d s0 e s0 /
e84 e e e d s0 e s0 /
g84 g g g g g /
b84d ot s0 a3 b s0 c84 b84 b b /
c83 c c c d s0 e s0 /

```

```
% Batt. 4
f83 f f f f f /
f84 f f a a a /
c85d ot s0 b3 c s0 f84 f f f /
b44 a8 c c c /
f83 f f f f f /
```

```
% Batt. 5
f43 gd b1 g /
r8 [ a84 d84 b d g ] /
d84 d d d d d /
c85 b1 a b8 b b b /
f43 gd b1 g /
```

```
% Batt. 6
e43 fd a1 f /
r8 [ g84 c84 a c f ] /
c84 c c c c c /
Gsxc b84 a1 g a8 a a a /
e43 fd a1 f /
```

```
% Batt. 7
b82 b b b bn b /
f84 f f f g g /
d84 d d d d d /
g14 f g a b a b c d8 f84 /
b82 b b b bn b /
```

```
% Batt. 8
c83 c c c c c Rb /
g84 c84 c c c c /
c44 e Gsxc f8 e1 f /
e84d f1 g4 Gsxb a8 g1 a /
c83 c c c c c /
```

Concerto per violoncello

Luigi Boccherini

Allegro

Cornì in Do

Vc. solo

Vl. 1 e 2

Viola

Vc. e Cb.

f

3

f

6

Musical score for system 6, measures 6-8. The system consists of six staves. The top staff is a single treble clef staff with a measure rest in measure 6, followed by two notes in measure 7 and a measure rest in measure 8. The second staff is a bass clef staff with a continuous eighth-note accompaniment. The third and fourth staves are grouped by a brace and contain piano accompaniment for the right and left hands, respectively, featuring sixteenth-note patterns. The fifth and sixth staves are also grouped by a brace and contain piano accompaniment for the right and left hands, respectively, with eighth-note patterns.

8

Musical score for system 8, measures 8-10. The system consists of six staves. The top staff is a single treble clef staff with a measure rest in measure 8, followed by two notes in measure 9 and a measure rest in measure 10. The second staff is a bass clef staff with a continuous eighth-note accompaniment. The third and fourth staves are grouped by a brace and contain piano accompaniment for the right and left hands, respectively, featuring sixteenth-note patterns. The fifth and sixth staves are also grouped by a brace and contain piano accompaniment for the right and left hands, respectively, with eighth-note patterns.

Largo

Vc. solo

Vl. 1 e 2

Viola

Vc. e Cb.

Vc. soli *Vc. e Cb. uniti*

4

7

Bibliografia

- [1] Musixtex mailing list. Lista di discussione dedicata alla stampa musicale con i programmi del gruppo **MusiX_TE_X**. Coordinatore: Maurizio Codogno.
- [2] Marc Baudoin. *Impara \LaTeX ! (... e mettilo da parte)*. <ftp://ftp.agm-ita.ensta.fr/pub/babafou/>, 1998. Traduzione italiana di *Apprens \LaTeX !* di Alessandro Cannarsi.
- [3] Andreas Egler Daniel Taupin, Ross Mitchell. *MusiX_TE_X. Using $T_{E}X$ to write polyphonic or instrumental music*. <http://icking-music-archive.org>. Manuale distribuito con MusiX_TE_X.
- [4] Michael Doob. *A Gentle Introduction to $T_{E}X$* . <http://www.loria.fr/services/tex/>, 1993. Ottima introduzione a $T_{E}X$. È reperibile anche presso tutti i siti CTAN (Combined $T_{E}X$ Archive Network), come <ftp.tex.ac.uk>.
- [5] Michael Doob. *Guida all'uso di $T_{E}X$* . http://www.ph.unito.it/~zaninett/index_italian.html, 1996. Traduzione italiana di [4].
- [6] Rainer Dunker. *Musixlyr—a MusiX_TE_X extension package for lyrics handling*. <http://icking-music-archive.org>. Manuale distribuito con Musixlyr.
- [7] Rainer Dunker. *Musixser—a MusiX_TE_X extension package for voice-wise music coding*. <http://icking-music-archive.org>. Manuale distribuito con Musixser.
- [8] Tobias Oetiker e altri. *Una (mica tanto) breve introduzione a $\LaTeX 2_{\epsilon}$* . <ftp://lorien.prato.linux.it/pub/guild/itlshort>, 1999. Traduzione italiana di *The Not So Short Introduction to $\LaTeX 2_{\epsilon}$* a cura di alcuni animatori del newsgroup it.comp.software.tex.
- [9] Guido Gonzato. *abcpp 1.4.3*. <http://abcplus.sourceforge.net/>, 2001 – 2003. Manuale d'uso accluso al programma.
- [10] Guido Gonzato. *Creare musica con ABC PLUS*. <http://abcplus.sourceforge.net/>, 2003 – 2005.

- [11] Donald Knuth. *The T_EXbook*. Addison-Wesley, Reading, Massachusetts, 1984. Il testo base per chi voglia approfondire ogni aspetto del sistema di tipocomposizione T_EX.
- [12] Leslie Lamport. *L^AT_EX: a document preparation system*. Reading, Massachusetts, 1984. L^AT_EX è la macro di T_EX più usata e può funzionare anche con MusiX_T_EX. Il presente scritto è stato redatto con L^AT_EX.
- [13] Leslie Lamport. *L^AT_EX 2_ε. The macro package for T_EX*. <http://www.loria.fr/services/tex/>, 1994. Guida dei comandi dell'ultima versione di L^AT_EX.
- [14] Dirk Laurie. *M-Tx: Music from Text*. <http://icking-music-archive.org>. Manuale distribuito con M-Tx.
- [15] Cornelius Noack. *Typesetting music with PMX*. <http://icking-music-archive.org>, 2008. Manuale in parte ricavato dalla presente opera.
- [16] Don Simons. *PMX. A preprocessor for MusiX_T_EX*. <http://icking-music-archive.org>. Manuale distribuito con PMX.
- [17] Daniel M. Zirin. *T_EX Made Easy. Using T_EX With The Plain Macro Package*. <http://www.loria.fr/services/tex/>, 1993. T_EX.

Indice analitico

- % , carattere di commento, [10](#), [19](#), [27](#), [75](#)
- ABC, [125](#), [126](#)
- ABC2MTeX, [84](#)
- ABCPP, [55](#), [125–129](#), [140](#), [141](#)
 - compilazione condizionata, [127](#)
 - macro, [126](#)
 - trasposizione, [128–131](#)
- accollatura, *vedi* sistema
- Bach, Johann Sebastian, [24](#), [49](#), [90](#), [107](#), [108](#), [124](#), [127](#), [128](#)
- batch file Windows, [19–21](#)
- Biebricher, Christof, [II](#), [IV](#), [99](#), [102](#)
- Boccherini, Luigi, [159](#), [162](#)
- Bononcini, Giovanni, [95](#)
- C, [12](#), [13](#)
- Caccini, Giulio, [140](#), [141](#), [159](#)
- Čajkovskij, Pëtr Il'ič, [52](#)
- CMackTeX, [2](#)
- Debussy, Claude, [32](#), [33](#), [87](#)
- Dufay, Guillaume, [115](#), [133](#), [136](#)
- Dunker, Rainer, [7](#), [24](#), [72](#)
- dvipdfm, [5](#), [10](#), [16–18](#)
- dvips, [5](#), [15–19](#)
- editor di testo, [1](#), [4](#), [10](#), [12](#), [16](#), [19](#)
- Egler, Andreas, [23](#)
- evince, [4](#), [15](#), [16](#)
- f2c, [11–13](#)
- Fink, [2](#), [4](#)
- Fortran, [11–13](#)
- fpc, [11](#), [13](#)
- g77, [13](#)
- gcc, [11](#), [13](#), [14](#)
- GhostScript, [4](#)
- Gonzato, Guido, [125](#)
- GSview, [4](#)
- gv, [4](#), [16](#)
- Handel, George Frederic, [108](#)
- Haydn, Franz Joseph, [27](#), [28](#)
- i-Installer, [2](#)
- Icking, Werner, [II](#), [24](#), [121](#)
- installazione, [3–14](#)
 - Linux, [9–14](#)
 - Windows, [5–8](#)
- Josquin des Prés, [96](#), [104](#)
- kdvi, [15](#), [16](#)
- KGhostView, [4](#), [16](#)
- Kneifl, Stanislav, [4](#), [24](#), [44](#)
- Knuth, Donald, [1](#)
- kpdf, [4](#)
- Laurie, Dirk, [23](#), [79](#)
- legature
 - bitmap, [19](#), [44](#), [90](#)
 - ps tipo K, [4](#), [7](#), [16](#), [19](#), [24](#), [45–48](#), [51](#), [61](#), [90](#)
 - ps tipo M, [V](#), [4](#), [8](#), [14](#), [17](#), [20](#), [24](#), [45–47](#), [69](#), [90](#)
- M-Tx, [III–V](#), [2–4](#), [9](#), [14](#), [17–20](#), [23](#), [72](#), [77](#), [79](#), [80](#), [82](#), [84–90](#), [93](#), [99](#), [101](#), [102](#), [104](#), [107](#), [111–113](#), [115](#), [119](#), [120](#), [126](#), [133](#), [140](#)
 - accordi, [79](#), [86](#), [111](#)
 - accordi per chitarra, [88](#), [89](#)
 - accordi, direzione dell'asta, [86](#)
 - aggiustamenti di posizione dei segni di dinamica e di espressione, [88](#), [89](#)
 - aggiustamenti di posizione del testo lirico, [93](#), [96](#), [102](#), [104](#)
 - armatura, [81](#)

- aste di battuta, 84, 85, 99
 aste di battuta fra i pentagrammi, 102, 104
 Bars/line, *vedi* battute per sistema
 battute per sistema, 81
 blocco, 81, 89, 102
 chiavi, 83, 84
 Choral, 83, 96, 104, 108, 111
 comandi PMX in linea, 80, 81, 84–87, 90, 93, 95, 96, 99, 100, 102, 104, 108, 115, 136, 141
 Composer, *vedi* titoli
 Continuo, 83
 corpo, 80, 84
 crescendo, 88–90
 diminuendo, *vedi* crescendo
 directory d'uscita, 111
 Disable, 81
 Enable, 81
 Flats, *vedi* armatura
 giustificazione verticale, 80, 81, 93, 96, 99, 102, 104, 107, 108, 120, 121, 136, 141
 Group, 83, 84
 gruppi ritmici, 83–86, 93, 111
 Indent, *vedi* rientro
 legature, 86, 93, 96, 98–101
 legature invisibili, 96
 legature tratteggiate, 101
 legature, segno di continuazione, 86, 101
 levare, 84, 85
 linee di battuta, *vedi* aste di battuta
 linee vocali, due per pentagramma, 107, 108
 linee vocali, sopra e sotto il pentagramma, 108, 111
 macro, 90
 melismi, 96
 Meter, *vedi* metro
 metro, 81, 85, 102
 metro 0/4, 99
 metro, cambio, 84, 85, 108
 misure incomplete, 84, 85
 Name, *vedi* titoli
 note, nomi, 111
 note, posizione, 86
 Octave, 81
 omissione delle aste di battuta, 84, 99
 Options, *vedi* opzioni di compilazione
 opzioni di compilazione, 80, 81, 93, 111, 112
 Pages, *vedi* pagine, numero
 pagine, numero, 81
 paragrafo lirico, 93, 95
 Part, *vedi* titoli
 pentagrammi, grandezza, 81, 82, 87, 90, 93, 95, 96, 99, 100, 102, 104, 107, 108, 136
 pentagrammi, numero, 82, 83
 PMX, *vedi* comandi PMX in linea
 Poet, *vedi* titoli
 preambolo, 79–85, 95, 102, 108, 111
 prepmx, 8, 13, 17
 raggruppamenti di note, *vedi* gruppi ritmici
 rientro, 81, 112
 ritornelli, 84, 85, 108
 segni di dinamica e di espressione, 79, 87–89
 Sharps, *vedi* armatura
 sillabazione, 95, 99–101, 108
 sillabazione individuale dei versi, 100, 101
 sinalefe, 95
 sistemi, numero, 81
 Size, *vedi* pentagrammi, grandezza
 Space, *vedi* giustificazione verticale
 stanghette, *vedi* aste di battuta
 Start, 80–82
 stile, 3, 81–83, 85–87, 90, 93, 95, 96, 99, 100, 102, 104, 107, 108, 136, 141
 Style, *vedi* stile
 Systems, *vedi* sistemi, numero
 testi, 87
 testo lirico, V, 79, 82, 88, 90, 93, 95, 108, 110
 Title, *vedi* titoli
 titoli, 81
 travature, *vedi* gruppi ritmici
 Uptext, 87, 89, 112
 Vocal, 83, 99, 111, 141

- vocalizzi, 93, 95, 98–100
- voci, numero, 85
- Macintosh, 2, 4
- metapost, 5, 8, 11, 17, 45
- MikTeX, 2, 4–6, 8, 15
- Mitchell, Ross, 23
- mktexlsr, 11
- Molekwa, Mthimkhulu, 114
- Mondrup, Christian, IV, 24
- Monteverdi, Claudio, 64
- Morimoto, Hiroaki, 4, 23, 24, 44
- Mozart, 2, 3, 81, 123
- mtex2ps.bat, 19, 20
- mtx2ps.bat, 19
- mtx2ps.sh, 21
- mtxm2pdf.bat, 20
- mtxm2pdf.sh, 22
- mtxm2ps.bat, 20
- mtxm2ps.sh, 22
- MusiCTeX, 23
- musixflx, 11, 15–18, 114
- musixlit, 98–100, 102, 104
- Musixlyr, 7, 24, 72, 79, 90, 93, 96, 100, 101, 115, 122
- musixpss, 8, 14, 17
- MusiXTeX, III, IV, 1–6, 11, 12, 15, 18, 19, 23, 24, 32, 43–45, 47, 51, 61, 68–70, 87, 90, 100, 106, 113–117, 120, 122
 - aspetto delle note, modifica, *vedi* definizioni
 - distribuzione, 4, 6, 7, 11, 24, 114
 - \endmuflex, 68, 114
 - \endpiece, 68
 - \enotes, 68, 69
 - \freqbarno, 70
 - giustificazione verticale, 120–122
 - \Interligne, 70, 96, 104, 108, 120, 124, 125, 134, 136, 141
 - \internote, 38, 39, 43, 44, 48, 52, 56, 57, 63, 89
 - \interstaff, 60, 115, 120, 121, 124, 125, 141, 159
 - \nobarnumbers, 2, 43, 49, 56, 69, 70, 87, 90, 93, 99, 100, 102
 - \normalbottom, 120
 - \notes, 2, 68, 69, 74
 - \noteskip, 41, 65
 - pedale, segno, 69
 - preambolo, 18
 - \raggedbottom, 120
 - \sepbarrules, 70, 114, 159
 - \setinterinstrument, 120, 121, 124, 125
 - \staffbotmarg, 120, 121
 - \stafftopmarg, 120
 - \startmuflex, 68, 69, 114
 - \startpiece, 68
 - \zcharnote, 69
- Noack, Cornelius, IV, 128, 129
- path, 5, 10, 19
- pdftex, 15, 17, 18
- PMX, III, IV, 2–4, 7, 8, 11, 13, 17–19, 23, 27–30, 32–34, 36–41, 43–45, 48, 49, 51, 53, 55–57, 59–69, 71, 72, 75, 79, 80, 82, 84–87, 89, 90, 101, 102, 104, 113–117, 119, 120, 123–128, 133
 - A, 59
 - A, giustificazione verticale, 60
 - A, grandezza delle alterazioni, 59
 - A, inclusione di file, 61
 - A, indicazione relativa delle alterazioni, 59
 - A, legature ps tipo K, 46–48
 - A, pentagrammi di diversa grandezza, 60
 - A, punti di valore, 59
 - A, spazio fra i pentagrammi, 60
 - accentato, 42
 - acciacature, 40, 41, 72, 74
 - accidenti, *vedi* note, alterazioni
 - accordi, 36, 39, 40
 - accordi, direzione dell’asta, 40
 - appoggiature, 40, 41, 72, 74
 - armatura, 30, 31, 55, 56
 - arpeggio, 53
 - aste di battuta, 51, 54
 - basso cifrato, *vedi* basso continuo
 - basso continuo, 64–66, 114
 - basso figurato, *vedi* basso continuo
 - basso numerato, *vedi* basso continuo

- battute, numero, 69, 70
 battute per sistema, 30
 blocco, 32–34, 41, 51, 54, 56–58, 62, 69, 71, 74, 76, 85
 cesura, 41, 42
 chiavi, 31, 32, 52, 53, 55, 74
 cifratura del basso, *vedi* basso continuo
 comandi TeX in linea, 60, 66, 68–70, 72, 74, 76, 77, 80, 89, 90, 93, 96, 100, 101, 106, 113, 114, 116, 119, 129, 159
 corona, 42
 corona inferiore, 42
 corpo, 27, 32, 44, 53, 66
 crescendo, 51, 52
 dimensioni della pagina, 62
 diminuendo, *vedi* crescendo
 directory d'uscita, 32
 direzione delle aste, 35, 48, 66
 estrazione delle parti, 39, 57, 59, 62, 75, 76, 127
 fusa, 123
 giustificazione verticale, 120–122, 159
 gruppi ritmici, 35, 48–51, 74
 impaginazione, 62, 63, 159
 indicazioni editoriali, 42
 legature, 43–48, 72, 74
 legature, nelle note ornamentali, 41
 levare, 30, 31, 64
 limiti, 72, 74, 75
 linee di battuta, *vedi* aste di battuta
 macro, 67, 68, 90
 metro, 27, 29, 30
 metro, cambio, 30, 54, 55, 64, 116
 midi, 35, 71–73, 76, 93, 127
 midi macro, 71, 72
 misure incomplete, 64
 mordente, 36, 42
 mordente inferiore, 42
 note ornamentali, *vedi* appoggiature e acciaccature
 note, alterazioni, 35, 36, 40, 41, 55, 56
 note, altezza, *vedi* note, ottava
 note, altezza delle aste, 38
 note, duine, *vedi* note, gruppi irregolari
 note, gruppi irregolari, 37, 38, 49, 60, 74, 75
 note, indicazione abbreviata del ritmo, 36, 39
 note, nomi, 33, 35, 39, 41, 56
 note, ottava, 34, 39, 41
 note, posizione, 35, 39
 note, punti di valore, 35, 36, 38
 note, terzine, *vedi* note, gruppi irregolari
 note, valore, 33, 34, 37, 38
 ornamenti, 41, 42, 72
 pagine, numero, 27, 30, 31, 58, 59, 62, 72, 76
 pause invisibili, 51
 pause, punti di valore, 39, 40
 pause, valore, 39
 pentagrammi, grandezza, 31, 76
 pentagrammi, numero, 29
 pizzicato, 42
 pmx.tex, 7
 pmxab, 7, 11, 12, 17
 preambolo, 27, 28, 31, 32, 52, 54, 55, 60, 62, 80, 114
 raggruppamenti di note, *vedi* gruppi ritmici
 respiro, 41, 42
 rientro, 31, 63
 ritornelli, 53, 54, 64, 71, 72
 scor2prt, 7, 11, 12, 39, 53, 55, 57, 59, 60, 62, 75–77, 122
 segni di dinamica e di espressione, 51, 52, 66
 segno, 42
 sforzando, 42
 simbolo di fine pentagramma, 32, 33
 simbolo di fine voce, 32, 33
 sistemi, numero, 30, 31, 62, 72, 76
 spazio minimo fra le note (W), 62
 spazio orizzontale (X), 61, 62
 staccato, 41, 42
 stanghette, *vedi* aste di battuta
 strumenti, cambio numero, 63
 strumenti, nomi, 31, 63
 strumenti, numero, 27, 29, 114
 tenuto, 42

- testi, [51](#), [57](#), [58](#)
- testo lirico, [72](#)
- titoli, [57](#), [58](#)
- tonalità, [30](#), [56](#), [57](#)
- tonalità, cambio, [55](#), [74](#), [85](#)
- trasposizione, [34](#), [55–57](#), [59](#), [66](#), [72](#), [102](#)
- travature, *vedi* gruppi ritmici
- trillo, [41](#), [42](#), [74](#)
- voci, numero, [29](#), [33](#), [124](#), [125](#)
- volta, [54](#), [72](#), [74](#)
- pmx2ps.bat, [19](#), [20](#)
- ps2pdf, [15–17](#)

- script Linux, [19](#), [21](#), [23](#)
- shell Linux, [10](#), [21](#)
- Simons, Don, [IV](#), [23](#), [24](#), [68](#), [117](#)
- sistema, [29](#), [31](#), [41](#), [53](#), [57](#), [60](#), [62](#), [63](#), [72](#), [74–76](#), [81–83](#), [112](#), [114](#)

- Taupin, Daniel, [23](#), [100](#)
- teTeX, [1](#), [2](#), [4](#)
- TeX, [1](#), [4–6](#), [9](#), [10](#), [13](#), [15–18](#), [28](#), [32](#), [45](#), [51](#), [63](#), [113](#), [115](#), [126](#)
 - albero locale, [4–6](#)
 - albero personale, [5](#), [9–11](#), [13](#)
 - albero principale, [4](#)
 - definizioni, [66](#), [70](#), [96](#), [100](#), [104](#), [106](#), [107](#), [115–119](#)
 - distribuzione, [1](#), [2](#), [4](#), [5](#), [8](#), [10](#), [15–17](#), [45](#)
 - estensione, [100](#), [102](#)
 - estensioni, [5](#), [6](#)
 - initexmf, [7](#)
 - kpsewhich, [10](#)
 - localtexmf, [5](#), [6](#)
 - macro, [1](#), [5](#), [6](#), [11](#)
 - texmf, [4](#), [10](#)
 - texmf.cnf, [10](#)
 - update-texmf, [10](#)
- texhash, [10](#), [11](#), [13](#), [14](#)
- TeXLive, [1](#), [4](#)

- uso, [15–23](#)

- Vivaldi, Antonio, [58](#), [136](#), [138](#), [140](#)
- voci, [32](#), [33](#), [36](#), [39](#), [53](#), [54](#), [59](#), [79–81](#), [83](#), [85](#), [89](#), [95](#), [107](#), [108](#), [111](#)
- Vogel, Oliver, [III](#), [IV](#), [124](#)
- Werner Icking Music Archive, [I](#), [III–V](#), [6](#), [8](#), [11](#), [13](#), [23](#), [24](#)
 - lista di discussione, [24](#)
- WYSIWYG, [1](#), [93](#)
- xdvi, [15](#), [16](#), [44](#)
- Yap, [15](#), [16](#), [44](#)
- Zipoli, Domenico, [116](#)