

# Zpravodaj Československého sdružení uživatelů TeXu

---

Karel Šebela

Sazba hudebních skladeb

*Zpravodaj Československého sdružení uživatelů TeXu*, Vol. 33 (2023), No. 1-2, 39–47

Persistent URL: <http://dml.cz/dmlcz/151757>

## Terms of use:

© Československé sdružení uživatelů TeXu, 2023

Institute of Mathematics of the Czech Academy of Sciences provides access to digitized documents strictly for personal use. Each copy of any part of this document must contain these *Terms of use*.



This document has been digitized, optimized for electronic delivery and stamped with digital signature within the project *DML-CZ*:  
*The Czech Digital Mathematics Library* <http://dml.cz>

$\text{\TeX}$  je užitečný nástroj pro sazbu textu, ale nemá v základu dobrou podporu pro sazbu skladeb. Na konferenci TUG 2022 zazněla přednáška o sazbě not, která porovnávala sazbu pomocí balíku *MusiX $\text{\TeX}$*  s nástroji *Musescore* a *Flat*, ale nezmínila preprocesory *PMX* a *M-Tx*, které sazbu zjednodušují. V tomto článku porovnám sazbu nástrojem *MusiX $\text{\TeX}$*  s jeho preprocesory a popíšu jejich užití. Dále rozeberu začlenění notových značek do textu odstavce. Po přečtení článku bude čtenář schopný vytvořit jednoduchý krátký úryvek hudebního díla a začlenit jej do  $\text{\TeX}$ ového dokumentu, či doplnit psaný text notovými značkami.

**Klíčová slova:** notový zápis, *MusiX $\text{\TeX}$* , *PMX*, *M-Tx*, *LilyPond*, *MuseScore*, *Flat*

## 1. Úvod

V tomto článku navazuji na prezentaci *Musical Composition Typesetting* představenou na konferenci TUG 2022 [1]. Samostatná prezentace sice zmiňovala některé nedostatky sazby konvenčními programy, ale celkově byla příliš stručná a mnohé principy vůbec nerozváděla. Rád bych v tomto článku prozkoumal možnosti, které  $\text{\TeX}$  v rámci sazby hudby umožňuje, a následně prodiskutoval jejich výhody a omezení.

Součástí tvorby hudebních děl je jejich zápis podle určitého systému. Takřka výlučně se dnes používá způsob zápisu formou not do notových osnov s pěti linkami. Tento způsob vychází již z počátku 11. století a přes svůj historický vývoj, například v počtu linek nebo trámcích mezi notami, se jeho podstata zásadně nezměnila [2].

Dříve manuální úpravy v současnosti usnadňují specializované editory. Na typografické úrovni řeší správné umístění not a rozestupů, sklony trámců, legat a dalších linek. Mimo jiné také zajišťují, že počet dob v každém z taktů souhlasí s uvedeným taktovým označením. Většina populárních programů (*MuseScore*, *Sibelius*, *Finale*, ...) umožňuje jednotlivé noty vkládat interaktivně přes WYSIWYG rozhraní. Jiný přístup využívá *LilyPond*, který, podobně jako  $\text{\TeX}$ , výsledné dokumenty sestavuje ze zdrojového kódu.

V sekci 2 shrnuji prezentaci *Musical Composition Typesetting* [1] z TUGu 2022 a popisují nedostatky představených nástrojů *MuseScore* a *Flat*. Následně v sekci 3 představuji sazbu pomocí balíku maker *MusiX $\text{\TeX}$*  a následně v sekci 4

představují preprocesory, které zápis v *MusiXTEXu* usnadňují. V sekci 5 představuji doplňkové L<sup>A</sup>T<sub>E</sub>Xové balíky, pomocí kterých lze sázet notové symboly v psaném textu. Článek uzavírám v sekci 6 popisem nedostatků *MusiXTEXu* a souvisejících preprocesorů a krátkým představením dávkového nástroje *LilyPond*, který je na T<sub>E</sub>Xu nezávislý a netrpí uvedenými nedostatky.

## 2. Motivace

Původní prezentace [1] připomínkuje nedostatky interaktivních editorů *MuseScore* a *Flat*. *MuseScore* je desktopová aplikace s otevřeným zdrojovým kódem, *Flat* je webová aplikace dostupná na doméně `flat.io`. Prezentace nezmiňuje, ve které aplikaci se vyskytují které problémy, připomínky jsem tak ověřoval v obou programech: v *MuseScore* verze 4.0.1 a ve zdarma dostupné edici *Flat*. Většinu chyb zmiňovaných v prezentaci se mi podařilo reprodukovat pouze v aplikaci *Flat*.

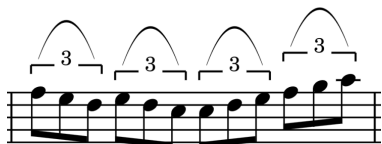
### 2.1. Nedostatky *MuseScore* a *Flat*

Připomínky k problémům výstupu sazby se týkají několika různých oblastí:

**Nastavení rozestupů not** Prezentace připomínkuje nemožnost volby vlastních rozestupů mezi jednotlivými notami. *Flat* nastavení rozestupů sice umožňuje, ale pouze pro celou skladbu a pouze v placené verzi [3]. Tyto rozestupy zůstávají stejné i po přidání textu písně, díky čemuž se jednotlivé slabiky překrývají. *MuseScore* těmito nedostatky netrpí.

**Sazba legat** *Flat* u not s trémci sází nadbytečné triolové svorky a legata nad nimi nereflktují informaci o výšce jednotlivých not. Výsledkem jsou nevzhledná legata s příliš prohnutým obloukem, uvedená na obrázku 1.

Dále jsou omezeny možnosti sazby vnořených legat, používaných u skladeb pro smyčcové nástroje. *Flat* umožňuje noty zároveň svázat pouze legatem a ligaturou. *MuseScore* umožňuje spojit legatem libovolné dvě noty a většinou i správně brání jejich kolizi. Příklad lehké kolize je uveden na obrázku 2.



Obrázek 1: Nevzhledná legata nad triolami v programu *Flat*. *MuseScore* sází trioly pod trémci bez svorek.



Obrázek 2: Kolize dvou legat v *MuseScore*. Legata se v tomto případě lehce dotýkají, nikdy se ale nepřekrývají.

**Export části skladby** *MuseScore* ani *Flat* neumožňují exportovat výběr dokumentu do grafického formátu [4]. Nelze tedy například vzít část skladby a zahrnout ji jako obrázek do jiného dokumentu; stránku je třeba ručně oříznout grafickým editorem.

### 3. Balík *MusiXTEX*

*MusiXTEX* obsahuje sadu maker a písem umožňujících sazbu hudby v systému  $\text{T}_\text{E}\text{X}$ . Zahrnuje symboly osnov, not, trámců, linek a dalších značek. *MusiXTEX* na základě posloupnosti  $\text{T}_\text{E}\text{X}$ ových maker určuje umístění jednotlivých prvků v rámci partitury. Na rozdíl od sazby lineárního textu umožňuje práci s několika notovými osnovami současně, což mimo jiné komplikuje řešení velikosti mezer mezi notami.

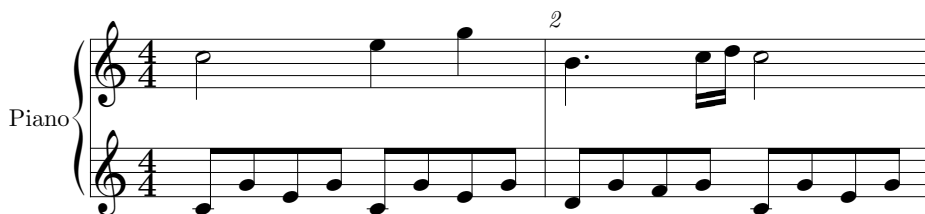
*MusiXTEX* kvůli řešení zalamování řádků zpracovává vstupní soubory třemi průchody. V prvním průchodu zpracuje vstupní soubor příkazem `etex` a poznačí si rozestupy taktů do souboru s příponou `.mx1`. Ve druhém průchodu procesor `musixflx` řeší optimální velikost rozestupů not a výstup zapisuje do souboru s příponou `.mx2`. Ten je následně zahrnut ve finálním průchodu `etexem`.

Zdrojový kód pro zahrnutí do  $\text{L}_\text{A}\text{T}_\text{E}\text{X}$ ového dokumentu, který po sestavení zobrazuje úryvek Mozartovy Sonáty C-dur, vypadá následovně [5]:

```
\begin{music}
\parindent10mm
\instrumentnumber{1}          % 1 nástroj
\setname1{Piano}              % pojmenovaný Piano
\setstafis1{2}                % se 2 notovými osnovami
\generalmeter{\meterfrac44}  % 4/4 taktové označení
\startextract                 % začátek úryvku
\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\tbl1\qb1k\en
\Notes\ibu0f0\qb0{cge}\tbu0\qb0g|\hl j\en
\zendextract                  % konec úryvku
\end{music}
```

V úryvku mezi příkazy `\startextract` a `\zendextract` se jednotlivé noty uvádí po řádcích mezi makry `\Notes` a `\en`. Nejprve je uvedena část not (ne nutně celý takt) pro spodní hlas (levou ruku), poté svislice `|` a následně odpovídající noty pro horní hlas (pravou ruku).

Vysázené dílo poté vypadá takto:



## 4. Preprocesory pro *MusiXTEX*

Popis not *MusiXTEX*ovými makry je poměrně zdlouhavý a u rozsáhlejších skladeb poněkud nečitelný. Namísto přímého psaní maker lze notový vstup psát ve zkráceném formátu a následně jej přeložit na prostý *TEX*ový soubor obsahující jednotlivá *MusiXTEX*ová makra. Konkrétně lze k tomu využít jeden z preprocesorů, buď preprocesor *PMX* nebo z něj vycházející procesor *M-Tx*.

### 4.1. Preprocesor *PMX*

Vstup preprocesoru *PMX* se zapisuje do odděleného souboru s příponou *.pmx*. *TEX*ový výstup lze generovat příkazem *pmxab*, snadnější je však zdroje sestavovat obalujícím luaskriptem *musixtex*, který ve výchozím chování dokument přímo vysází do formátu *.pdf*.

Výslednou partituru ve formátu *ps* nebo *pdf* pak lze použít samostatně nebo zahrnout do jiného dokumentu *L<sup>A</sup>T<sub>E</sub>X*ovým příkazem *\includegraphics* jako vektorovou grafiku. Protože má však vysázená skladba rozměry celé strany, je žádoucí ji před zahrnutím oříznout – buď ruční specifikací přes možnost *trim*, nebo automaticky například externím nástrojem *pdfcrop*.

Preamble vstupního souboru *.pmx* je velmi minimální. Začíná posloupností 12 čísel, které určují parametry celé partitury. Popisují například počet notových osnov, nástrojů, definice taktových označení nebo počet stran. Následuje seznam jmen nástrojů a určení klíčů pro jednotlivé osnovy. Poslední řádek preamble tvoří adresářová cesta, do které má být zapsán *TEX*ový výstup [6]. Celá preamble včetně komentářů může vypadat například takto:

```
% počet osnov, nástrojů, 4/4 takt, označený C, délka předtaktí,
      2          1          4 4          0 6          0
% předznamenání, počet stran, řádků, velikost, přesazení
      0          1          1          20          0.07
Piano
```

```
bt
./
```

Tělo vstupního souboru *PMX* poté určuje výšky a délky jednotlivých not. Noty se zapisují písmeny **a–g**, následovanými číslicemi určujícími délku (0 **o**, 2 **l**, 4 **l**, 8 **l**, 1 **l**, 3 **l**) a oktávu. Oddělení taktů svislou čarou není povinné, ale usnadňuje kontrolu chyb. Dalšími příkazy lze přidávat posuvky (**s #, f b, n b**), dynamiku a další značky. Oproti balíku *MusixTeX* je většina příkazů zkrácena na jeden znak, což zmíněný úryvek Sonáty C-dur zkracuje na pouhé dva řádky (zarovnání mezerami zde nehraje roli):

```
c83 g83 e g c83 g83 e g | c83 g83 f g c83 g83 e g /
c25 e4 g4 | bd- c1 d1 c2 /
```

Délku not i oktávu si preprocesor udržuje podle předchozí noty, lze je tedy místy vynechat. U větších intervalových skoků je nutné uvést konkrétní oktávu absolutně číslem nebo relativně symboly **+/-**. Poněkud neintuitivní je u *PMX* zápis hlasů v obráceném pořadí – levý hlas klavíru je v tomto příkladu popsán horním řádkem.

## 4.2. Preprocesor *M-Tx*

Ve starších verzích procesor *PMX* nepodporoval sazbu textů písní. V současné verzi lze texty písní v *PMX* psát v uvozovkách, které jsou poté obaleny do *TeX*ového makra **pmxlyr**, dále rozvíjeného na *MusiXTeX*ové příkazy.

Alternativní rozhraní pro sazbu textů nabízí preprocesor *M-Tx* (z anglického „Music from Text“) [7]. Formát souboru *M-Tx* je založen na formátu *PMX*; podstatněji se liší ve formátu preambule, kde místo seznamu hodnot používá seznam příkazů ve formátu **Klíč: hodnota**. Vstupní soubory se zpracovávají luaskriptem **musixtex**, který postupně volá jednotlivé procesory. Soubor je nejprve přeložen do formátu **.pmx**, poté zpracován preprocesorem *PMX* a vysázen *TeX*em. Oproti *PMX* se zde jednotlivé osnovy zapisují v přirozeném vertikálním pořadí.

S ohledem na český text mi přišlo žádoucí zkusit sazbu českých písní včetně diakritiky. Bohužel, písmena s diakritikou nelze do *MusiXTeX*ových zdrojových souborů jednoduše zadávat. Částečné řešení, které umožňuje psát text s diakritikou v UTF-8, jsem přejal z dokumentace *MusiXTeX*u [5, sekce 23.4]. Spočívá ve změně kategorie písmen s diakritikou a následném překladu *LuaTeX*ovým backendem. Takto lze text psát přirozenou češtinou:

```
%\catcode`\á\active \defá{\'{a}}
%\catcode`\č\active \defč{\v{c}}
%\catcode`\ě\active \defě{\v{e}}
%\catcode`\ř\active \defř{\v{r}}
```

```

%%\catcode`\í\active \defí{\'\{i}}
%%\catcode`\Š\active \defŠ{\v{S}}
%%\catcode`\ť\active \defť{\v{t}}
%%\catcode`\ž\active \defž{\v{z}}
Title: Bože, cos ráčil
Composer: V. Kment
Poet: V. Šťastný
Sharps: 1
Space: 7
Style: Singer Organ
Organ: Voices RH LH,Ped; Clefs G F; Continuo
Width: 150mm

```

```

a a8 g f4 f | f8 f e f a4 g |
L: 1. Bo-že, cos rá-čil před ti-sí-ci ro-ky
L: 2. Na ve-le-hra-dě brat-ři ze So-lu-ně,
L: 3. Ja-zy-kem rod-ným Bo-ží chvá-lu pě-li,
f e8 ( d d4 ) e | ds c8 d f4 e |
d+ c8 b a2 | b2 c4n b |
d2d c4n | [ b8 a+ ] g8 f e2 |

```

LuaTeXový backend lze zvolit zadáním příslušného argumentu:

```
musixtex -F "luatex --output-format=pdf" soubor.mtx
```

Vysázená skladba poté vypadá následovně:

## Bože, cos ráčil

V. Šťastný

V. Kment

1. Bo - že, cos rá - čil před ti - sí - ci ro - ky  
2. Na ve - le - hra - dě brat - ři ze So - lu - ně,  
3. Ja - zy - kem rod - ným Bo - ží chvá - lu pě - li,

## 5. Notové značky uvnitř textu

Mimo sazbu not do notových osnov lze i psaný text doplnit notovými symboly, například pro ilustraci hudební teorie. Symboly, které používá *MusiXTEX*, mají při použití v textu nesprávnou velikost a nelze je tak do textu jednoduše začlenit.

Umístění hudebních značek do odstavců nabízí balíky *musicography* a *lilyglyphs*. Balík *musicography* má omezenější výběr, navíc jeho makra pro taktová označení kolidují s těmi *MusiXTEX*ovými.

Balík *lilyglyphs* využívá k sazbě notových symbolů glyfy písma Emmentaler z programu *LilyPond*. K vyhledání jednotlivých glyfů přitom závisí na balíku *fontspec*, díky čemuž vyžaduje k překladu Lua<sup>A</sup>T<sub>E</sub>X. Pro porovnání vzhledu notových značek z obou balíků vizte tabulku 1.

Noty ♪ ♫ ♬ ♧,	posuvky # ♭ ♮,	taktová označení $\frac{3}{4}$ $\frac{4}{4}$ $\frac{6}{8}$ <b>C</b> <b>♩</b> .
Noty ♪ ♫ ♬ ♧,	posuvky # ♭ ♮,	taktová označení $\frac{3}{4}$ $\frac{4}{4}$ $\frac{6}{8}$ <b>C</b> <b>♩</b> .

Tabulka 1: Porovnání vzhledu a rozestupu některých symbolů z balíků *musicography* (nahore) a *lilyglyphs* (dole) při oddělování mezerami `\thinspace = \,`. Vysázené prvky z jednotlivých balíků mají odlišné mezery, taktová označení *musicography* mají pouze základní vzhled.

Zahrnutí balíku *lilyglyphs* konkrétně u šablony *csbulletin* zahlásí chybu při duplicitní definici `centerbox`. Jako workaround lze makro `centerbox` před načtením balíku *lilyglyphs* oddefinovat příkazem `\let\centerbox\relax`. Zároveň je u této šablony první z taktových označení *lilyglyphs* mírně nezarovnané, konkrétní příčinu jsem již ale nedohledal.

## 6. Nedostatky T<sub>E</sub>Xových systémů

V porovnání s konvenčními WYSIWYG editory jsou současné T<sub>E</sub>Xové systémy poměrně obtížné. Začínající uživatele může odradit nedostatek zdrojů, často omezený na oficiální dokumentace, které jsou často příliš dlouhé nebo neobsahují dostatek názorných příkladů. Názornější mohou být příklady skladeb a dodatečné příručky dostupné z webu *Werner Icking Music Archive* [8], konkrétně například návod *Typesetting Music With PMX* [9].

Dalším nedostatkem T<sub>E</sub>Xových systémů a jejich preprocesorů jsou místy nečitelné chybové zprávy. Jako příklad uvádím zprávu, která může uživatele zaskočit při zpracování souboru *PMX* s nesprávným formátem hlavičky. Hlášení v tomto případě nevypisuje kód *PMX*, ale přímo formátovací funkce Fortranu, ve kterém je *PMX* implementován [6]. Každopádně hledání příčiny problému příliš neusnadňuje:



```
fmt: read unexpected character
apparent state: internal I/O
last format: (f1.0)
lately reading sequential formatted internal I/O
```

Mimoto je také možné, že samotná podstata textové sazby nemusí být pro řadu uživatelů zcela intuitivní, což může bránit širšímu rozšíření těchto nástrojů.

## 6.1. Program LilyPond

Alternativu k  $\text{T}_{\text{E}}\text{X}$ ovým makro balíkům může představovat program *LilyPond*. Na rozdíl od představených procesorů není vázán na  $\text{T}_{\text{E}}\text{X}$  a existují k němu grafické editory, například *Frescobaldi* či *Denemo*. Zatímco *Frescobaldi* ze zadaného zdrojového kódu generuje náhled skladby, *Denemo* umožňuje naopak noty vkládat interaktivně a zpětně generuje kód pro LilyPond [10].

Kód LilyPondu lze zahrnout i do  $\text{T}_{\text{E}}\text{X}$ ových souborů, konkrétně buďto inline prostřednictvím makra `lilypond` nebo častěji přes stejnojmenné prostředí:

```
\begin{lilypond}[quote,fragment,staffsize=26]
  c' d' e' f' g'2 g'2
\end{lilypond}
```

Pro překlad dokumentu lze použít makrobalík *LyLuaT<sub>E</sub>X* krátce představený v minulém čísle Zpravodaje [11, sekce 5]. Po zahrnutí příkazem `\usepackage{ly-luatex}` lze sázet dokument přímo příkazem `lualatex`:

```
lualatex --shell-escape document.tex
```

Vysázený úryvek poté vypadá takto. Pokud je cílem uvést úryvek do jiného dokumentu (jako zde) a využívá se pro ořez `pdfcrop`, je potřeba vypnout  $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ ové číslování stran příkazem `\pagenumbering{gobble}`.



## Odkazy

1. PARK, Christopher; PARK, Emily. *Musical composition typesetting* [online]. 2022-07-24. [cit. 2022-11-12]. Dostupné z: <https://youtu.be/bwbk2xiMLSg>.
2. PATERSON, Jim. *A Short History of Musical Notation* [online]. [cit. 2023-03-08]. Dostupné z: <https://www.mfiles.co.uk/music-notation-history.htm>.

3. TUTTEO LIMITED. *Flat: Number of measures per system and notes spacing* [online]. [cit. 2023-03-13]. Dostupné z: <https://flat.io/help/en/music-notation-software/layout-measures>.
4. WEISS, Isaac; SABATELLA, Marc. *Export Selection File menu option* [online]. 2014-12-26. [cit. 2022-11-21]. Dostupné z: <https://musescore.org/en/node/42336>. Feature request.
5. VOGEL, Oliver et al. *MusixT<sub>E</sub>X: Using T<sub>E</sub>X to write polyphonic or instrumental music* [online]. 2021-08-29. Ver. 1.35 [cit. 2022-11-21]. Dostupné z Package documentation na <https://ctan.org/pkg/musixtex>.
6. SIMONS, Don. *PMX: a Preprocessor for MusiX<sub>T</sub><sub>E</sub>X* [online]. 2022-02-04. Ver. 2.98 [cit. 2022-11-22]. Dostupné z Package manual na <https://ctan.org/pkg/pmx>.
7. LAURIE, Dirk. *M-Tx: Music from Text* [online]. 2019-01-15. Ver. 0.63c [cit. 2022-11-21]. Dostupné z Package documentation na <https://ctan.org/pkg/m-tx>.
8. *MusiX<sub>T</sub><sub>E</sub>X and related software: Introduction* [online]. [cit. 2023-03-13]. Dostupné z: <https://www.icking-music-archive.org/software/htdocs/Introduction.html>.
9. NOACK, Cornelius C. *Typesetting music with PMX* [online]. 2013-05. Ver. 2.821 [cit. 2023-03-13]. Dostupné z: <http://icking-music-archive.org/software/pmx/pmxccn.pdf>.
10. *Front-end Applications* [online]. [cit. 2023-03-13]. Dostupné z: <https://lilypond.org/easier-editing.html>.
11. NOVOTNÝ, Vít. Vysokoúrovňové jazyky pro T<sub>E</sub>X. *Zpravodaj ČSTUGu*. 2022, roč. 32, č. 1–4, s. 35–48. Dostupné z DOI: 10.5300/2022-1-4/35.

## Summary: Musical Composition Typesetting

T<sub>E</sub>X is a useful tool for text typesetting; however, it doesn't feature a good support for composition typesetting at its core. The TUG 2022 conference featured a talk on notation typesetting which compared the *MusiX<sub>T</sub><sub>E</sub>X* package with the *MuseScore* and *Flat* tools, but did not mention the *PMX* and *M-Tx* preprocessors. In this article, I compare typesetting using *MusiX<sub>T</sub><sub>E</sub>X* and its preprocessors and describe its usage. In addition, I describe the incorporation of note symbols into a paragraph text. After reading the article, the reader will be able to create a short simple excerpt of a piece of music and incorporate it into a T<sub>E</sub>X document, as well as add musical symbols to a written text.

**Keywords:** sheet music, music engraving, *MusiX<sub>T</sub><sub>E</sub>X*, *PMX*, *M-Tx*, *LilyPond*, *MuseScore*, *Flat*

Karel Šebela, 514509@mail.muni.cz