

# Zpravodaj Československého sdružení uživatelů TeXu

---

Zdeněk Wagner

Konverze Word Perfect = > LaTeX

*Zpravodaj Československého sdružení uživatelů TeXu*, Vol. 3 (1993), No. 3, 135–138

Persistent URL: <http://dml.cz/dmlcz/149681>

## Terms of use:

© Československé sdružení uživatelů TeXu, 1993

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>

### 3.1 Under Plain DOS (without DPMI)

Set EMX to EMX.EXE:

```
set emx=c:\bin\emx.exe
```

It would be a~good idea to place the above line in your AUTOEXEC.BAT. Also add the following line to your AUTOEXEC.BAT:

```
set emxopt=-t  
set emxtmp=c:/temp
```

under the assumption that you have the C:\TEMP directory.

### 3.2 Under DPMI (including Windows' DOS Box)

You have to set EMX to RSX.EXE:

```
set emx=c:\bin\rsx.exe
```

#### [4] Summary

The method presented here is summarized as follows:

1. Remove the DOS extender attached to tex386.exe and mf386.exe
2. Attach a~DOS extender \*loader\* to tex386.exe and mf386.exe
3. Get DOS extender for plain DOS and for DPMI.
4. Properly set DOS extenders which the DOS extender loader loads when you run tex386.exe or mf386.exe

Good luck.

Young

---

---

## Konverze Word Perfect $\Rightarrow$ L<sup>A</sup>T<sub>E</sub>X

---

ZDENĚK WAGNER

Když člověk jednou propadne T<sub>E</sub>Xu (L<sup>A</sup>T<sub>E</sub>Xu, A<sub>M</sub>S-T<sub>E</sub>Xu, ...), říká si, že konverze jsou zbytečné. Nejjednodušší je napsat vše rovnou v T<sub>E</sub>Xu. S tímto názorem souhlasím, jenže život není žádná humanita. Často se

stává, že získáme soubor, který někdo napsal nějakým textovým editorem, a my jej máme vysázet v  $\text{\TeX}$ u. V tomto článku bych se chtěl podělit o své zkušenosti (a trápení) s konverzí českých textů z Word Perfectu 5.1 do  $\text{\LaTeX}$ u.

K této problematice jsem se dostal tehdy, když jsem získal české texty z jistého semináře na VŠE psané ve Word Perfectu. Věděl jsem z FAQ,<sup>1)</sup> že existuje konverzní program **WP2LATEX**. Původně jej napsal v Turbo Pascalu R. C. Houtegen z TUE-Eindhoven. Proto jsou některé komentáře psány holandsky. Později jej pomocí **p2c** zkonvertoval Glenn Geers ze Sydney University do jazyka C a poté jej modifikoval Michael Covington z University of Georgia. Jak se dalo předpokládat, program neumí konvertovat české znaky. Dlouhé samohlásky byly nahrazeny správnými příkazy jako `\'e`, neboť se vyskytují např. ve francouzštině. Většina českých znaků se však zkonvertovala na `\P`, což není moc užitečné. Než ale popíšu, jak jsem problém vyřešil, budu pokračovat v líčení dalších potíží.

Z dokumentu z VŠE jsem si tedy zjistil kódování češtiny ve Word Perfectu 5.1. Dlužno poznamenat, že to nebyl ani kód Kamenických, ani Latin2, ani KOI8CS. Původně jsem chtěl svoji verzi dát k dispozici na nějaký server, ale text z VŠE neobsahoval všechna česká písmena a konverzní funkce tedy nebyla kompletní. Proto jsem od svého záměru upustil.

Po krátkém čase mi volal známý, že v jistém pražském nakladatelství mu tisknou cosi napsané českým Word Perfectem 5.1<sup>2)</sup> a bohužel se jim porouchala tiskárna. Hledá tedy někoho, kdo by mu to nějak vytiskl. Proto jsem navštívil jinou známou, která pro jiné nakladatelství psala texty českým Word Perfectem 5.1. Napsal jsem si u ní na disketu všechna česká písmenka, abych mohl dokončit konverzní program. K mému velkému zděšení měl její český Word Perfect 5.1 úplně jiné kódování češtiny než Word Perfect 5.1 z VŠE. Nyní vás asi nepřekvapí, že známý, pro kterého jsem si konverzní program připravoval, měl ve své verzi Word Perfectu 5.1 kódování zcela odlišné. . .

Ideu obecného konverzního programu jsem tedy vzdal. Z programu **WP2LATEX** jsem vytvořil pomocný nástroj, s nímž lze konverzi provést, avšak definici kódování je nutno upravit podle konkrétního dokumentu.

---

<sup>1)</sup> Máte-li přístup na INTERNET, víte, že zkratka FAQ znamená „Frequently Asked Questions“ a příslušný dokument obsahuje nejčastěji kladené otázky týkající se  $\text{\TeX}$ u a samozřejmě též odpovědi.

<sup>2)</sup> Všimněte si, že stále mluvím o „stejně“ verzi Word Perfectu!

Ted' už se konečně dostaneme k výkladu, jak je vše uděláno. Word Perfect má řadu znakových sad (např. znaková sada 8 obsahuje řeckou abecedu). Konverzní program tedy musí rozeznat nejenom kód znaku, ale i sadu. Pro tento účel si WP2LATEX vytváří tabulku. Ta je definována ve funkci:

```
Local Void Ext_chr_init(LINK)
```

Funkce obsahuje řadu příkazů. Pro ilustraci uvedu následující:

```
LINK->char_set[0x1c] = 0x4;
LINK->char_code[0x1c] = 0xb;
strcpy(LINK->ext_lat[0x1c], "\\pounds ");
```

Hodnota 0x1c nemá žádný hlubší význam, je to pouze index v konverzní tabulce. Uvedené příkazy specifikují, že znak ze sady 0x4 s kódem 0xb se nahradí řetězcem `\\pounds`, tj. vytiskne se jako £. Pokud se nějaký znak v tabulce nenajde, v L<sup>A</sup>T<sub>E</sub>Xovém souboru se objeví \P. Ošetření tohoto případu se vyskytuje ve funkci

```
Local Void Character(LINK)
```

v následujících příkazech:

```
if (found)
    strcpy(ch, LINK->ext_lat[j - 0x80]);
else
    strcpy(ch, "\\P");
```

V programu jsem udělal dvě úpravy. Nejprve jsem zrušil všechny příkazy ve funkci `Ext_chr_init`. Abych dostal něco užitečnějšího než strohé \P, nahradil jsem poslední příkaz výše uvedeného příkazu `if` příkazem:

```
sprintf(ch, "\\P{%x/%x}", (unsigned)char_set, (unsigned)char_code);
```

Tím získám ve výpisu jak číslo znakové sady, tak kód nepoznaného znaku. Kdyby se v textu vyskytoval £ a nebyl v tabulce, měl bych ve výstupním souboru `\P{0x4/0xb}`. Po dohodě s autorem textu nebo často i intuicí lze kódování rozluštit a doplnit do funkce `Ext_chr_init`. Do konverzní tabulky lze zadat až 256 znaků.

WP2LATEX má svůj vlastní „style option“ pro tisk L<sup>A</sup>T<sub>E</sub>Xem. Nezkoušel jsem konvertovat matematiku ani tabulky, takže nevím, jak se v těchto případech konverzní program zachová. Při konverzi hladkých textů doplňuje určitá svá makra, ale naštěstí je jich tak málo, že je lze v libovolném editoru snadno ručně odstranit a dokument pak lze snadno vytisknout i v plain T<sub>E</sub>Xu.

*Zdeněk Wagner*  
wagner@csearn.bitnet  
wagner@earn.cvut.cz

---

---

## Několik poznámek z tiskařské praxe

---

Jiří TLACH

Každý, kdo se T<sub>E</sub>Xem nějaký čas zabývá, dostane dříve či později za úkol vysázet tiskové podklady pro nějakou brožuru, skripta, časopis, manuál... nebo knížku. Protože vždy a všichni šetří – tedy co nej-laciněji a samozřejmě v T<sub>E</sub>Xovské kvalitě.

Pro nejběžnější ofsetový tisk je nutno vyrobit (jako předlohu) na transparentní matérii zrcadlově převrácenou stránku (tzv. špígl). S Mat-tesovými ovladači to jde poměrně snadno (v `options` použijeme transformaci strany s parametrem `tr4`, `5`, `6` nebo `7`. Měli bychom však mít na paměti některá technická omezení vyplývající z konstrukce nejčastěji používaných laserových tiskáren a hlavně téměř magickou číslici 300 DPI, charakterizující jejich běžné rozlišení (jehličkové k tomuto účelu nemá smysl používat a užití inkoustových nutno odzkoušet na konkrétních materiálech).

Celý proces poté funguje tak, že se v tiskárně předlohy smontují na větší formát (obvykle A3 nebo A2) a kontaktně přepokopírují na kovolist (zde je písmo normálně čitelné a po vyvolání přijímá barvu, narozdíl od ostatního povrchu kovolistu, který barvu nepřijímá). Na kovolist se nášší barva, gumovým válečkem se přenáší na papír – a vesele tiskneme. Samozřejmě trochu zjednodušuji, ale pro pochopení principu mi to budíž prominuto.