

# DTP v systému L<sup>A</sup>T<sub>E</sub>X

## úvodní informace

RNDr. Tomáš Mikulenka

výukový materiál v rámci grantového projektu *Beznákladové ICT pro učitele*

Kroměříž, březen 2012



# Vznik a historie T<sub>E</sub>Xu

Autor T<sub>E</sub>Xu – **Donald Erwin Knuth**, profesor Stanfordské univerzity

- 1977 – nespokojenost Knutha se sazbou knihy *The Art of Computer Programming* byla impulsem k vývoji nového sázecího systému
- 1978 – první verze (označována jako T<sub>E</sub>X 78)
- 1980 – založena organizace TUG (T<sub>E</sub>X Users Group)
- 1990 – vznik CSTUG (československé sdružení TUG)
- 1992 – ukončení vývoje T<sub>E</sub>Xu, jen opravy případných chyb; současná verze – 3,14159

# Co je T<sub>E</sub>X

- autorský systém pro sazbu odborného textu
- volně šiřitelný (tzv. public domain)
- programovatelný (má vlastní makrojazyk)
- dávkový: výsledný vzhled dokumentu ovlivňujeme zápisem v textových blocích – dávkách
- dobře dokumentovaný (vyšel knižně)
- výstup je nezávislý na zařízení

# Co je L<sup>A</sup>T<sub>E</sub>X

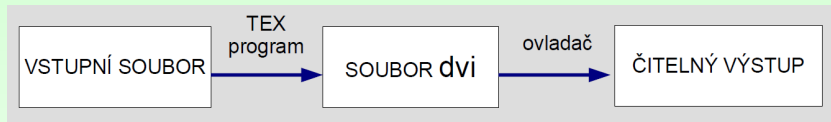
- často využívaný formát T<sub>E</sub>Xu (volně šiřitelná nadstavba)
- vytvořený Leslie Lamportem (80. léta 20. stol., Digital Equipment Corporation)
- základní myšlenka: zpřístupnit složitý jazyk T<sub>E</sub>Xu běžným uživatelům (typografickým neprofesionálům) a usnadnit jim precizní vysázení požadovaného textu
- čteme [la-tech] – Lamportův T<sub>E</sub>X

# Kde pořídít program

Lze např. zdarma stáhnout instalaci pro OS Win ze stránek projektu MikTeX Christiana Schenka ([www.miktex.org](http://www.miktex.org)).

The screenshot shows the MikTeX website homepage. At the top left, it says "please support the project: [give back](#)". To the right is the large blue MikTeX logo. Below the logo, there is a navigation menu on the left with "Home", "About" (highlighted in blue), "MiKTeX Packages", "MiKTeX Portable", "MiKTeX Tools", "Download" (highlighted in blue), and "MiKTeX 2.9". The main content area on the right contains the text: "Welcome to the MikTeX project page!", "New here? [Learn more](#) about MikTeX...", and "Want to support the project? Please [give back](#)!". Below this is a "Recent News" section with a bullet point: "■ [MiKTeX on Windows 8](#) (Thu, 15 Sep 2011)".

# Práce v L<sup>A</sup>T<sub>E</sub>Xu – schema



Práce se podobá programování – probíhá v těchto fázích:

- 1 příprava (editace) zdrojového dokumentu
- 2 překlad – vysázení
- 3 prohlížení, kontrola

# Práce v L<sup>A</sup>T<sub>E</sub>Xu

Tento sled kroků je třeba opakovat tak dlouho, dokud nedosáhneme požadovaného vzhledu dokumentu.

**Nevýhoda** tohoto způsobu práce: výsledek nevidíme IHNEDE (ve zdrojovém textu), což odradí zejména začátečníky.

**Výhoda:** efektivním zápisem několika potřebných příkazů se systém T<sub>E</sub>X (L<sup>A</sup>T<sub>E</sub>X) sám postará o precizní a bezchybné zpracování a vysázení

# Příprava textového dokumentu

Protože L<sup>A</sup>T<sub>E</sub>X není žádný WYSIWYG editor, musíme do dokumentu kromě vlastního textu zahrnout ještě **příkazy pro způsob sazby**. Jsou to:

- jeden aktivní znak (např. `&`, `$`, `^`)
- posloupnost `\z` (`z` je neabecední znak, např. `\#`, `\\`, `\%`, `\@`)
- posloupnost `\slovo` (`slovo` je posloupnost písmen, např. `\small`, `\uv`, `\alpha`, `\textit`, ...)



# Příprava dokumentu – závorky

Některé příkazy mají parametry uváděné v závorkách:

$\{$       $[$       $($

- $\{$  povinné parametry – bez těchto závorek příkaz platí jen pro první znak
- $[$  nepovinné parametry – lze je vynechat včetně závorek
- $($  (jen u příkazů pro kreslení obrázků)

# Struktura dokumentu

Každý dokument určený ke zpracování v L<sup>A</sup>T<sub>E</sub>Xu má tuto rámcovou strukturu:

```
\documentclass[volby]{třída}[datum vytvoření]
    ... preamble ...
\begin{document}
    ... textová část ...
\end{document}
```

# Třída dokumentu \documentclass

Parametr {třída} definuje styl, jakým má být dokument vysázen:

- 1 letter ... dopis
- 2 report ... zpráva
- 3 article ... článek
- 4 book ..... kniha
- 5 slides ... fólie k promítání

Samostatnou třídu tvoří Beamer – tvorba prezentací.

# Používání balíčků `\usepackage`

V preambuli se deklaruje použití tzv. balíčků. Ty mohou měnit funkci určitých příkazů nebo definovat nové příkazy pro přidání dalších funkcí.

Balíček = sada příkazů L<sup>A</sup>T<sub>E</sub>Xu v souboru s příponou `.sty`; připojuje se příkazem `\usepackage`. Jediným příkazem lze načíst více balíčků.

```
\usepackage{color}
\usepackage{graphics,ifthen,amstex}
\usepackage[czech]{babel}
```

# Skupina a prostředí

**Skupina** = úsek textu ohraničený složenými závorkami.

**Prostředí** = úsek textu ohraničený příkazy

```
\begin{název} ... \end{název}
```

kde `název` je slovo s definovaným významem.

Text uvnitř skupiny či prostředí se zpracuje jiným způsobem než ostatní text a to v závislosti na příkazu ve skupině nebo na parametrech prostředí.

Písmo 12 pt. { \Large A písmo zvětšené. }

```
\begin{center} text na střed \end{center}
```

Skupiny a prostředí se mohou do sebe libovolně vnořovat, ale nesmějí se křížit.

# První dokument

Minimální kostra dokumentu L<sup>A</sup>T<sub>E</sub>Xu:

```
\documentclass[12pt,a4paper]{article}
\usepackage[czech]{babel}
\usepackage[utf8]{inputenc}
\begin{document}
  vlastní text – test české diakritiky:
  Příliš žluťoučký kůň úpěl ďábelské ódy.
\end{document}
```

# Odstavce

Přechod na nový odstavec lze v L<sup>A</sup>T<sub>E</sub>Xu provést dvěma způsoby:

- a) vynecháním (aspoň jednoho) řádku
- b) příkazem `\par` (odstavec = paragraph)

Přechod na nový řádek bez odstavcového zlomu:

- a) příkazem `\\`
- b) příkazem `\\ [2 mm]` (parametr řádkování)

# Mezery v textu

- mezislovní mezera: sází se mezerníkem
- nezlomitelná mezera: znak tilda (~)  
příklad: v~lese, z~pole, o~tom, s~nimi
- zúžená mezera: \, (chová se jako nezlomitelná mezera)  
příklad: J.\,S.\,Bach, 30\,cm
- rozšířená mezera (k oddělení matematických výrazů nebo určitých celků): \quad – šířka 1 em; \qquad – šířka 2 em
- mezera libovolné délky: \hspace{8.4 mm} – vodorovná, \vspace{2.5 ex} – svislá



# Dělení slov

L<sup>A</sup>T<sub>E</sub>X provádí dělení slov automaticky s využitím slovníku výjimek. Pokud by mělo dojít k chybnému rozdělení slova, lze nařídit jiné dělení příkazem `\-`.

Při zápisu `olympiá\-`da se dané slovo rozdělí buď jako „olympiá-da“ nebo vůbec.

Při používání problematických slov lze na začátek textové části napsat příkaz `\hyphenation{seznam}`, např.

```
\hyphenation{orga-ni-zační při-bliž-ně cvrnkl}
```

Slovo „cvrnkl“ pak nebude rozděleno nikde v celém textu.

# Stupeň písma

Stupeň (size) rozhoduje o rozměrech písma. V L<sup>A</sup>T<sub>E</sub>Xu se používá relativní stupnice velikostí (jednotlivé příkazy – viz tabulka na další straně).

Základní velikost (`\normalsize` – vyznačena červeně) je určena třídou dokumentu, ostatní jsou pak od ní odvozeny:

# Přehled velikosti (stupně) písma

příkaz L <sup>A</sup> T <sub>E</sub> Xu	VELIKOST PÍSMO
<code>\tiny</code>	nejmenší písmo
<code>\scriptsize</code>	velikost horních/dolních indexů
<code>\footnotesize</code>	velikost poznámek pod čarou
<code>\small</code>	text dokumentu
<code>\normalsize</code>	<b>text dokumentu</b>
<code>\large</code>	text dokumentu
<code>\Large</code>	text dokumentu
<code>\LARGE</code>	text dokumentu
<code>\huge</code>	text dokumentu
<code>\Huge</code>	text dokumentu

## Atributy písma definované uživatelem

Následující sekvencí příkazů může uživatel definovat použité písmo v dokumentu individuálně:

```
\fontfamily{rodina}  
\fontseries{váha}  
\fontsize{velikost}{řádkování}  
\selectfont
```

---

**Příklad:** `\fontfamily{ppl}\fontseries{b}`  
`\fontsize{20}{24}\selectfont`

nastaví písmo Palatino, tučné, stupeň 20 pt, řádkování 24 pt.