Úvod do TEXu

Přednáška je zaměřena pro začínající uživatele Linuxu, kteří právě objevili, že v Unixových systémech je obvyklé formátovat dokumenty prostřednictvím TEXu, ale ještě nevědí, co to přesně je. Budou nastíněny základní principy TEXu, vysvětlena terminologie a zmíněny odkazy na další dokumenty. Za příznivých okolností (konstelace hvězd a funkčnost techniky) může být předveden TEX „živě" při úpravě jednoduchého dokumentu.

O programu

TEX se program na přípravu elektronické sazby. Název TEX je odvozen od počátečních písmen slova „technologie", což v řečtině prý znamená technologii i umění. Proto se název čte řecky „tech" ($\tau\varepsilon\chi$).

Program byl napsán profesorem Stanfordovy univerzity D. E. Knuthem koncem sedmdesátých let. Autor publikoval algoritmy programu k volnému použití ve speciálním jazyce WEB společně s nástroji, které konvertují zápis algoritmů do standardního Pascalu. To umožnilo program implementovat na libovolném operačním systému, na kterém běžel překladač Pascalu.

Nezávislost programu na výpočetním prostředí (architektura počítače, operační systém, výstupní zařízení) je jeho velkou předností. Mezi další přednosti patří:

- existence dokonalé dokumentace
- silný jazyk interního interpretru
- velmi dobrý návrh mnoha interních algoritmů pro zpracování sazby
- možnost bezplatného použití programu i jeho částí třeba ke komerčním účelům
- existence pevného bodu při přípravě dokumentů.

K poslednímu bodu: poslední verze TEXu z roku 1989 je autorem fixována a nebude do budoucna dále vyvíjena. Dokument, který jste od tohoto data připravili nebo připravíte, bude TEXem zpracovatelný se zcela stejným výsledkem třeba i za sto let.

Kde začít

Dobrým startovním bodem při pátrání po rozs8hl7ch informacích k programu může být `www.cstug.cz`. CSTUG je (podobně jako CZLUG) sdružení uživatelů programu, kteří svými členskými příspěvky podporují rozvoj informovanosti o programu a případný vývoj souvisejícího software. Dalším vhodným startovním bodem při pronikání do možností programu je dokument „První setkání s TEXem", který lze najít například na `ftp://math.feld.cvut.cz/pub/cstex/doc/prvni.tar.gz`.

Různé implementace

Na implementacích programu ze zdrojových textů v jazyce WEB pro jednotlivé operační systémy už nepracuje sám autor, ale mnoho dalších programátorů po celém světě. Mimo volně šířených implementací se můžeme setkat též s komerčními implementacemi, protože licence k programu je natolik volná, že to umožňuje. Mezi nejznámější implementace patří web2c, která má licenci GPL a jejím autorem je Karl Berry. Jedná se o implementaci použitelnou pro libovolný systém, pro který existuje rozumný překladač jazyka C. Mezi přímé následníky této implementace dnes patří teTeX (pro Unixy) a fpTeX (pro MS Windows). Kromě implementace programu TEX samotného a dalších spolupracujících programů pro danou počítačovou platformu je potřeba zabalit stovky maker, fontů a dalšího software a připravit vše ke snadné instalaci. Příkladem takové distribuce může být CDčko TEXlive, které připravuje každoročně TUG (sdružení uživatelů pracující v USA).

Dávkový způsob zpracování

TEX zpracovává dokumenty dávkovým způsobem. Je potřeba připravit do souboru vstupní text dokumentu společně se značkami, které vyznačují (nejlépe) logickou strukturu dokumentu nebo (hůře) přímo vyznačují nějaký požadavek na výsledný vzhled tisku. Kromě toho se TEXu předloží k načtení sada maker, které převádějí logické značky v dokumentu na příkazy určující konečný vzhled tiskoviny. Tato makra se dají dodatečně modifikovat a tím upravovat výstup podle konkrétního návrhu typografa. Pokud soubor s textem dokumentu je dobře strukturován logickými značkami, nemusí sazeč při závěrečné úpravě do tohoto souboru vůbec zasahovat. Stačí mu jen modifikovat TEXová makra, která jsou rovněž zapsána v textové podobě většinou v pomocných souborech.

Výstupem z TEXu je binární soubor s příponou `dvi`, který obsahuje přesný a definitivní popis rozmístění znaků na všech stranách dokumentu. Datový formát je samozřejmě nezávislý na platformě a na výstupním zařízení. Jsou tam uvedeny odkazy na fonty, přičemž o realizaci kreseb jednotlivých znaků se postarají až tzv. `dvi`-ovladače. Právě tyto programy musejí mít přístup k fontům a vytvářejí výstup závislý na použitém výstupním zařízení. Například program `dvips` převádí `dvi` do PostScriptu a při této činnosti zavádí do výstupu použité fonty, například ve formátu Type1. Prohlížeč TEXového výstupu, program `xdvi`, je vlastně také `dvi`-ovladač, který realizuje vzhled dokumentu v okénku X Window Systému. Pokud `dvi` soubor odkazuje na PostScriptové fonty, ovladač si na pozadí požádá Ghostscript o jejich vyrastrování.

Teoretická dělba práce

Jak bylo řečeno, TEX načítá při zpracování dokumentu makra, která definují chování TEXu v místě použitých logických značek v textu a určují tím konečný vzhled dokumentu. Makra tedy ovlivňují zpracování dokumentu ve dvou rovinách. Jednak je jimi určeno, jak mají vypadat logické značky v textu a jednak definují typografii dokumentu. To umožňuje určitou dělbu práce uživatelů TEXu. Odborník spolupracuje s typografem a navrhuje makra TEXu. Na druhé straně autor dokumentu si od odborníka vyzvedne seznam několika málo značek, kterými vyznačí logickou strukturu svého dokumentu. Text zapisuje v obyčejném editoru bez formátovacích vymožeností. Zajímá jej jen obsah dokumentu a starost o formu svěří odborníkovi.

Teoreticky by měl odborník nejprve navrhnout logické členění a z toho vyplývající značkování dokumentu a pak nechá autora dokument napsat (nebo písařku dokument přepsat z rukopisu). Dále před zpracováním TEXem musí odborník rozhodnout, jakou zvolí typografii dokumentu a podle těchto požadavků napíše pro TEX makra navazující na použité logické značky. Pak teprve dokument zpracuje TEXem, přičemž výsledný vzhled, který si promítne například prostřednictvím programu `xdvi` do vedlejšího okénka, může dodatečně upravovat v textovém editoru modifikací svých maker.

Takto „teoreticky" se to ale dost často nedělá. Existují hotová makra pro TEX, která na jedné straně deklarují způsob značkování dokumentu a na druhé straně definují výchozí vzhled výstupu, který je možno více či méně komfortně dále modifikovat úpravou hodnot některých parametrů maker nebo vnitřních registrů TEXu. Snahou je, aby autor mohl svůj dokument sepsat pokud možno sám bez pomoci odborníka a bez hlubší znalosti poměrně složitého makrojazyka TEXu. Mezi nejjednodušší sadu maker patří plain, dílo samotného autora TEXu. Tato sada maker nezavazuje uživatele ani odborníka ke konkrétnímu značkování dokumentu, ale poskytuje výborné startovní pole pro vytváření vlastních maker, která se o základní makra z plainu mohou opírat.

Mezi hodně používanou výchozí sadu maker patří LATEX, dílo Leslie Lamporta, které dnes dále zvelebují lidé z tzv. LATEX teamu (`www.latex-project.org`). Toto makro dává autorovi technické literatury do rukou hotový návrh logického značkování dokumentu a dále předpřipravené šablony pro výsledný vzhled – tzv. styly. Říkáme, že píšeme v LATEXu, pokud jsme přijali způsob značkování dokumentu popsaný v Lamportově manuálu „LATEX– A Document Preparation System“. Naopak, říkáme že píšeme v plainu, pokud jsme se rozhodli deklarovat vlastní značkování a připravit pro každý dokument vlastní makra na míru opírající se pouze o základní makra plainu.

TEX je schopen načtená makra uložit do binárních souborů, tzv. formátů, pro další použití. Při zpracování dokumentu pak TEX už nenačítá textovou verzi maker znova, ale velmi rychle si obnoví své znalosti z těchto binárních formátů. V TEXových instalacích najdeme většinou dva nejčastěji používané formáty: plain a LATEX. Fázi načítání textové verze těchto maker a ukládání do binární podoby nazýváme inicializací formátu. Pak už stačí jen formát opakovaně používat na různé dokumenty. I při takovém použití formátu lze TEXu dodatečně nabídnout další makra, která původní makra z formátu předefinují nebo definují nová jen pro daný konkrétní dokument.

Čeština v TEXu

Čeština a slovenština pro TEX je k dispozici v balíku CSTEX. Balík zahrnuje doplňující makra pro plain (tzv csplain) a pro LATEX (tzv. cslatex). Dále jsou zde vzory dělení slov pro český a slovenský jazyk. Je zde také sada výchozích textových fontů, tzv. CSfonty. Ty obsahují úplnou českou abecedu a vznikly modifikací defaultní rodiny fontů pro TEX Computer Modern. Kromě těchto fontů jsou zde nástroje umožňující přímo použít základní sadu 35 fontů obsažených v každém PostScriptovém RIPu včetně české a slovenské abecedy. CSTEX najdete na `http://math.feld.cvut.cz/olsak/cstex`.

Kromě balíčku CSTEX se snaží češtinu řešit multijazykový modul TEXových maker pro LATEX zvaný Babel. Dnes jej najdete skoro v každé distribuci TEXu. Bohužel, tento modul není kompatibilní s cslatexem, takže uživatelé LATEXu musejí být opatrní a musejí si vybrat, jakým způsobem chtějí ve svých dokumentech češtinu používat.

Uživatelské rozhraní

Začátečníci se často ptají na uživatelský komfort při použití TEXu. Mají rovněž tendenci srovnávat TEX s jinými programy na přípravu elektronické sazby jen z pohledu uživatelského rozhraní. Zapomínají ale na to, že uživatelské rozhraní není vlastností TEXu samotného, ale je dáno několika na TEXu nezávislými faktory:

- uživatelský komfort použitého textového editoru
- grafické prostředí (to závisí na použitém operačním systému)
- použitá sada značek v dokumentu (závisí na zvoleném makru nebo odborníkovi)

- možnost vyhledat, jaká se má použít logická značka (implementace nápovědy)

Ve snaze přiblížit uživatelům zvyklým na interaktivní zpracování textu (chci kus textu kurzívou a chci hned teď vidět, jak ta kurzíva bude vypadat, jsem ochotem kvůli tomu bloudit a klikat myší, nejsem ochoten si zapamatovat, že stačí napsat „`\it`“) vznikají programy, které se z uživatelského hlediska podobají Wordu, ale při povelu k uložení dokumentu se vytvoří textový soubor standardně značkovaný podle LaTeXu. Příkladem takového projektu je program LyX.

O jednom mýtu zvaném WYSIWYG

Často lze slyšet nesprávný názor, že TeX není WYSIWYG. Doslovný překlad zkratky „co vidíš to dostaneš“ nás ale opravňuje tento názor vyvrátit. Za použití `dvi`-prohlížeče, který je samozřejmou součástí každé TeXové distribuce, můžeme na obrazovce skutečně vidět přesně to, co dostaneme při případném tisku na papír. Dokonce můžeme tento vzhled „za chodu“ měnit tak, že upravujeme TeXová makra ve vedlejším okénku textového editoru a na pozadí spouštíme opakovaně TeX.

Autoři myšlenky, že TeX není WYSIWYG, měli zřejmě na mysli, že TeX neumožňuje interaktivně zasahovat do vzhledu dokumentu v tom okénku, kde je výsledný vzhled zobrazen. Skutečně, dloubáním myší do okénka s programem `xdvi` nemůžeme zpětně ovlivnit obsah dokumentu a jeho maker, tedy TeXové vstupy. Proto se hodí spíše mluvit o dávkovém zpracování kontra interaktivním a zkratku WYSIWYG do toho raději netahat.

Následníci TeXu

Poté, co Knuth uzavřel projekt TeX a nedovolí nikomu program bez změny názvu měnit, vznikly některé další projekty, které navazují na algoritmy z TeXu a nějakým způsobem jej rozšiřují. Mezi nejznámější a zřejmě nejpoužívanější rozšíření patří pdfTeX, který umožní vygenerovat místo `dvi` přímo formát PDF. Podrobněji si o pdfTeXu můžeme přečíst na `www.cstug.cz/pdftex/index.html`.

Můžeme se také setkat s projektem ΩTeXu (implementuje vnitřně 16bitové kódování). Na zcela novém následníku TeXu pracuje skupina pod zkratkou NTS (New Typesetting System). Výstupem její činnosti je zatím mírně rozšířený TeX nazvaný e-TeX (rozšíření na úrovni jazyka WEB). Také vytvářejí kompletní přepis TeXu do objektově orientovaného jazyka Java.