



# OBSAH

Zdeněk Wagner: Úvodník . . . . .	177
Vít Zýka: Příprava dokumentů pro formátování . . . . .	178
Vít Zýka: Článek <code>CONTEXtEm</code> : tutoriál . . . . .	200
Vít Zýka: Logo <code>CONTEXtEm</code> : tutoriál . . . . .	209
Petr Březina: Zrcadlová sazba . . . . .	212
Petr Březina: Sazba trojjazyčné knihy . . . . .	227
Vít Zýka: Postřehy ze setkání <code>TEX</code> perience 2008 . . . . .	237
Vít Zýka: Postřehy ze setkání <code>CONTEXt</code> meeting 2008 . . . . .	241
První oznámení: Konference: <code>TEX</code> perience 2009 . . . . .	243
Euro <code>TEX</code> 2009 & 3rd <code>CONTEXt</code> Meeting . . . . .	245
TUG 2009 . . . . .	247

Součástí tohoto čísla je DVD `TEX` Collection 2008.

Zpravodaj Československého sdružení uživatelů `TEX`u je vydáván v tištěné podobě a distribuován zdarma členům sdružení. Po uplynutí dvanácti měsíců od tištěného vydání je poskytován v elektronické podobě (PDF) ve veřejně přístupném archívu dostupném přes <http://www.cstug.cz/>.

Své příspěvky do Zpravodaje můžete zasílat v elektronické podobě, nejlépe jako jeden archivní soubor (`.zip`, `.arj`, `.tar.gz`). Postupujte podle instrukcí, které najdete na stránce <http://bulletin.cstug.cz/>. Pokud nemáte přístup na Internet, můžete zaslat příspěvek na disketě na adresu:

Zdeněk Wagner  
Vínohradská 114  
130 00 Praha 3

Disketu formátujte nejlépe pro DOS, formáty Macintosh 1.44 MB a EXT2 jsou též přijatelné. Nezapomeňte přiložit všechny soubory, které dokument načítá (s výjimkou standardních součástí `CTEX`u), zejména v případě, kdy vás nelze kontaktovat e-mailem.

ISSN 1211-6661 (tištěná verze)

ISSN 1213-8185 (online verze)

Vážení členové, dovoluji si Vás přivítat v letošním posledním čísle Zpravodaje v zastoupení předsedy, Jaromíra Kubena, který je v době přípravy pracovně v zahraničí.

Dobrou zprávou je, že se podařilo nejen naplnit letošní čtyři čísla Zpravodaje, ale poprvé po několika letech nevzniká ve vydávání skluz. Patří za to dík Pavlu Strížovi, který při přípravě letošních Zpravodajů významně pomohl, ale i laskavosti a vstřícnosti nakladatelství KONVOJ, s. r. o., kde doslova lámou časové rekordy. A v neposlední řadě musím poděkovat všem autorům, kteří se rozhodli, že se na těchto stránkách podělí o své zkušenosti, neboť bez jejich článků, tutoriálů a reportáží by Zpravodaj vůbec nemohl existovat.

Toto číslo obsahuje též články, jež se nevešly do již tak přerostlého konferenčního čísla. Druhým důvodem přesunu bylo opoždění vývoje  $\text{T}_{\text{E}}\text{X}$  Live 2008, takže nebyla naděje, že DVD bude hotovo v termínu konání  $\text{T}_{\text{E}}\text{X}$ perience, kdy jsme již chtěli dát Zpravodaj č. 3 účastníkům. Vydání samostatného čísla 4, obsahujícího DVD  $\text{T}_{\text{E}}\text{X}$  Live, se tedy jevila jako příznivější varianta než rozšíření konferenčního sborníku na dvojčíslo.

DVD  $\text{T}_{\text{E}}\text{X}$  Collection 2008 je součástí tohoto čísla. V letošní distribuci se objevuje několik přelomových změn. Hlavní novinkou je zcela přepracovaná infrastruktura, která umožňuje nejen instalaci, ale též aktualizaci po internetu, vlastnost, kterou znají uživatelé  $\text{MiK}_{\text{T}}\text{E}_{\text{X}}$ u, ale v  $\text{T}_{\text{E}}\text{X}$  Live dosud chyběla. Druhou významnou změnou je to, že se podařilo zahrnout nové moduly pro češtinu a slovenštinu, jejichž autorem je Petr Tesařík, do standardní distribuce babelu. Odtud se dostaly do  $\text{T}_{\text{E}}\text{X}$  Live. Výsledek grantu, o němž jste si mohli přečíst ve Zpravodaji 1/2007, již tedy není teorií ani záplatou, kterou si lidé musí instalovat sami, ale standardní součástí distribuce. V  $\text{T}_{\text{E}}\text{X}$  Live 2008 též najdete novou třídu používanou pro články Zpravodaje. Výčet novinek tím samozřejmě zdaleka nekončí. Podrobnosti si můžete přečíst v dokumentaci na DVD. A ti, kdo navštívili valnou hromadu, si vše mohli vyslechnout z úst nejpovolanejších, z přednášky Norberta Preininga, jednoho z tvůrců této populární distribuce.

Na závěr mi již nezbyvá než popřát Vám příjemné čtení, mnoho  $\text{T}_{\text{E}}\text{X}$ ových i osobních úspěchů v nadcházejícím roce a těšit se na setkání na konferenci  $\text{T}_{\text{E}}\text{X}$ perience 2009, o níž se též můžete dozvědět z tohoto čísla Zpravodaje.

*Zdeněk Wagner*  
*zpravodaj@cstug.cz*

V tomto článku formulujeme obecné principy dobrého dokumentu a klademe požadavky na jeho editaci, zpracování a vizualizaci. Na základě těchto požadavků ukazujeme, že vhodným formátem je strukturně označovaný dokument. Vysvětlujeme strukturní značkování a popisujeme jeho vlastnosti. Na závěr uvádíme nástroje pro práci se strukturně značkováným dokumentem a ukazujeme způsoby jeho formátování T<sub>E</sub>Xem.

## 1. Cíle, definice

*Cílem dokumentu je předat informaci*<sup>1</sup>. Protože dobře vysázený dokument ke snadnému pochopení a nezkreslenému podání informace nabádá, není důležitý jen jeho obsah, ale i forma. Proto je dobré při tvorbě dokumentu věnovat pozornost jeho formátování.

Aby k předání mohlo dojít, je třeba ji podávat *srozumitelnou formou* a přihlídnout ke způsobilostem cílového čtenáře. Takový čtenář má svá očekávání, která nejsou-li naplněna, bývá zklamán či zmaten. V takovém případě hrozí, že autorovy myšlenky nebudou pochopeny, nebo že čtenář ani dokument nedočte. Pravda, najdeme i úspěšné práce, které toto pravidlo očividně porušují, viz obr. 1, 2. I ty však splňují čtenářovo očekávání. Jde o očekávání nonkonformity, novátorství či zábavy a ne o předání obsahové informace, jak je tomu ve většině našich případů. Informace je nejlépe vstřebávána, proudí-li přirozeně a bez formálních rušivých vlivů.

Abychom se vyhnuli spekulacím o platnosti zde nabízených pravidel a doporučení, vymezíme oblast našeho zájmu. Za *dokument* budeme v tomto článku považovat:

- periodicky vydávaný/aktualizovaný/sériový text dané struktury (zákon, předpis, nařízení, podniková zpráva, memorandum, konferenční příspěvek),
- rozsáhlý text jednotné struktury (softwarová dokumentace, encyklopedie),
- text využitelný k mnoha různým účelům (podnikový adresář, informace o obcích, statistická data).

Pojem *formátování* definujeme jako proces, který vtiskne datům formu vhodnou ke vnímání člověkem. Ačkoliv tato forma může být zvuková (hlasová syntéza) či hmatová (Brailovo písmo), budeme se věnovat té nejběžnější: vizuální – tiště-

---

<sup>1</sup>Informaci zde chápeme v nejobecnějším smyslu, včetně vizuální i estetické.





Obrázek 2: Celostránkové dřevorezy Josefa Váchala jsou těžko čitelné svou kostrbatou formou, archaickým jazykem i obsahem. Přesto jsou dnes velmi ceněny (a to nejen peněžitě). Důvodem není jen výjimečná řemeslná i umělecká kvalita a nadčasový obsah. Formální stránka dodává dílu složitě se deroucímú na svět věrohodnosti. Podtrhává osobnostní komplikovanost a existenční obtíže autora, který se pochopení svého díla nedožil. *Josef Váchal, Vidění sedmery dnů a planet. Paseka 1998.*

## 2. Principy dobrého dokumentu

Philip Taylor ve svém článku [26] formuloval tři principy dobrého dokumentu. Připojíme-li k nim ještě čtvrtý, pak dobrý dokument charakterizuje:

1. informace;
2. struktura;
3. jednotnost;
4. tradice, estetika.

### 2.1. Informace

Tvůrčí svoboda a lehkost internetového publikování (textu i obrazu) násobí tzv. informační explozi. Možnost snadno veřejně vyjadřovat svůj názor je bezesporným kladem, na druhé straně však vede k zaplavení zajímavých, nezkradených a komplexních zpráv. Mělo by proto být ctí autora přinášet pouze nová data prositá od těch méně kvalitních. A mít na paměti, že dokument bez přidané hodnoty nemá cenu tvořit!

Když už dokument vytvoříme, je potřeba věnovat pozornost kontrole dat. Je nutné, aby autor text po sobě několikrát přečetl. Hodně užitečné práce zastanou programy na kontrolu překlepu a pravopisu. Přesto je nezbytné, aby text četl ještě někdo jiný, nejlépe školený *korektor* znalý oboru. Korektury se dělají na papírové verzi. Odhalí se tam mnohem více chyb a označí se snadněji než na obrazovce.

Informace je téměř výhradně věcí autora, zatímco úkolem sazeče je formátování. Ačkoliv obě činnosti lze provádět nezávisle, je dobré je koordinovat. Některé úpravy dokumentu totiž sazeč již nemůže ovlivnit, pokud na ně autor při jeho vytváření nemyslel. Pokud by např. autor tohoto textu popsal výše uvedené požadavky na formu dokumentu v odstavci – namísto přehledných a stručných odrážek, použije nejen mnoho slov navíc, ale čtenáři se tyto informace budou i hůře vyhledávat. Podobným příkladem je nepoužití tabulky či obrázku. Nejčastější příčinou nevhodné struktury ‚rukopisu‘ není neznalost náležitých prvků, ale jejich špatná dostupnost. Autor by proto měl být před vytvářením dokumentu poučen o možných a vhodných elementech pro plánovaný druh textu i se způsoby jejich vytváření a vkládání.

## 2.2. Struktura

*Struktura* (pochází z latinského *struere*, skládat, sestavovat, budovat, pořádat) označuje způsob složení, vnitřního uspořádání nějakého objektu, zejména pokud vykazují nějaké pravidelnosti a zákonitosti. Je to souhrn vztahů mezi prvky nějakého seskupení [Wikipedie].

Struktura dokumentu je souhrn jeho vizuálně upravených obsahových celků a jejich vztahů. Obsahové celky nazýváme *strukturní* (logické, vnitřní, obsahové)<sup>2</sup> *elementy* a realizujeme je pomocí *elementů grafických* (vnějších, formátovacích, formálních). Funkcí grafických elementů je vzájemně oddělovat a odlišovat elementy strukturní a umožnit tak čtenáři se v dokumentu orientovat.

### 2.2.1. Strukturní elementy

Každý dokument se může skládat z mnoha různých strukturních elementů závislých na typu dat. Jiné elementy bude obsahovat adresář a jiné kniha. Námí definovaný dokument může obsahovat tyto obecné části a podčásti tvořící hierarchickou strukturu [21]:

- dokument  $\subset$  patitíl, protitíl, titulní list, vydavatelský záznam, anotace, tiráž, abstrakt, věnování, předmluva, mezitíl, obsah, literatura, rejstříky, oddíly, kapitoly, podkapitoly;
- oddíl, kapitola, podkapitola, ...  $\subset$  nadpis, odstavce, obrázky, tabulky, algoritmy i s jejich popisky;
- odstavec  $\subset$  poznámky pod čarou, marginálie, písmena, symboly, matematické výrazy, zvýraznění, odrážky číslované i nečíslované, definice, tabulky, obrázky, akce, hypertextový link.

Čím obecnější je definice logické struktury dokumentu, tím je obtížněji použitelná. To platí zvláště pro autory, kteří nerozlišují mezi strukturními a grafickými

---

<sup>2</sup>Možná by bylo lepší používat termín *logické elementy*, abychom zdůraznili významovou složku nad grafickou. Budeme se však držet převládajícího názvosloví.

částmi. Proto je užitečné pro každý dokument vytvořit vlastní definici struktury dokumentu používající jen potřebné elementy a tyto pojmenovat přiléhavými názvy. Toto je typická práce redaktora dokumentu.

### 2.2.2. Grafické elementy

Grafické elementy můžeme rozdělit na viditelné a neviditelné.

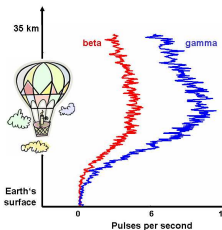
**Viditelné elementy** dokumentu jsou vcelku intuitivní grafické prvky. Vyjmenujme ty nejvýznamnější, s jejich atributy a typickými vnořeními viditelnými elementy:

- kniha, brožura, . . . , článek (médiu, stránkové zrcadlo)  $\subset$  stránky, sloupce;
- stránka (velikost, řádkový rejstřík)  $\subset$  sloupce, záhlaví, zápatí, stránková číslice, marginálie, plovoucí objekty včetně jejich popisků;
- sloupec (pořadí, šířka)  $\subset$  řádky, rámečky, linky;
- řádka (šíře, výplň)  $\subset$  znaky, linky, rámečky, obrázky;
- znak (písmo, velikost, barva, atribut);
- linka (tloušťka, počátek, konec);
- rámeček (okraj, šířka, výška);
- obrázek (grafický formát, velikost);
- akce a hypertextový link.

## I METEOROLOGICAL EXPERIMENT

Analyzed data come from Prague-Libuš upper air meteorological station of the Czech Hydrometeorological Institute, where every month the vertical profiles of beta and gamma radioactivity are measured by the radioactivity sonde system. The radioactivity sensor consisting of two Geiger-Müller gamma and beta tubes is a part of meteorological balloons which ascends from the earth's surface up to 35 km and detects short current pulses coming from the interaction between the radiation and the tube wall material.

After several "re-calculations" the complicated measurement process results into the data pairs  $(x_i, y_i)_{i=1}^n$ , where  $x$  represents the altitude and  $y$  the **average number of pulses per second** in the fixed altitude  $x$ . Let us note that  $y$  is proportional to the **radiation intensity**.



## II APPLIED REGRESSION MODELS

The first aim of the analysis was to suggest a parametric regression model  $Y_i = m(X_i) + \varepsilon_i$ , where  $m(\cdot)$  represents mean amount of radiation and  $\varepsilon_i$  a random "error" term. As described by Hlubinka (2004), the models  $m(\cdot)$  are based on **Richards growth curve**  $R(x)$  and its derivative  $r(x)$ . Unfortunately, Richards curve itself does not describe properly the measurements in low atmospheric layers. That's why we propose two extended additive models. The first model consists of  $r(x)$  and a **simple linear function**, meanwhile the second model of  $r(x)$  and **logistic growth curve** being of the form

Obrázek 3: Příklad špatného použití mezer při formátování titulku. Velikost mezer by měla odpovídat logickému vztahu sousedních viditelných elementů. Proto před titulkem má být mezer větší než ta pod ním. Oddělíme tak kapitoly od sebe a zároveň svážeme titulek s tělem kapitoly. Výřez.



**Neviditelné elementy** bývají často opomíjeny [26, 25]. Jde totiž o bílá místa na stránce. Přitom tyto elementy hrají rozhodující roli v oddělení jednotlivých viditelných elementů, i v chápání správných vztahů mezi nimi. Tak např. text **Toto není nadpis!**

budeme chápat jako příkaz sumarizující předchozí odstavec a ne jako titulek další kapitoly. Rozhodující pro náš úsudek je vertikální odsazení, viz obr. 3. Mezi neviditelné elementy řadíme

- Okraje stránky – jejich funkcí je, aby prsty nepřekážely čtení při držení dokumentu, a bylo snadné vyhledat stránkovou číslici (vhodné místo je při vnějším okraji na všech stranách bez výjimky), identifikovat záhlaví. Komponování stránky a jejich okrajů je popsáno např. [2, 1].
- Mezery okolo textových elementů – oddělují elementy od sebe. Příkladem je vyznačení odstavce (obr. 4), již zmíněné mezerování titulků kapitol, obrázků ap., oddělení plovoucích elementů a jejich titulků od hlavního textu, oddělení záhlaví a zápatí.
- Řádkování – mělo by využívat hodnoty, které navrhl tvůrce použitého písma, optimalizované pro čitelnost a neslití řádků.

tomu musí být u horké sazby dodnes. Dává mnohem více volnosti v tvaru i návaznosti, než jí měl tradiční písmař. A tato tvůrčí přednost se již u některých novinek projevila.

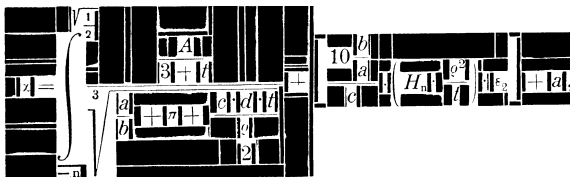
Johannes Gutenberg, když se v jeho rukou zrodil zázrak knihtisku, mohl ztěžší předpokládat směr vývoje v naší době. Věřil ve význam svého vynálezu a uvědomoval si „v bázni boží i ve vsí skromnosti“ jeho dosah, i když se musel trápit přizemnějsími starostmi, např. jak vyrovná zadlužení, jímž by hradil svůj objev. Ale více než sto padesát let po něm zhodnotil Gutenbergův čin jiný světoobčan, Čech, Jan Amos Komenský. Ve své „Živé tiskárně“ píše prorocky kromě jiného: „*Nejkrásnějším darem božím je vynález tiskových liter, jimiž se knihy nesmírně rychle rozmnožují. Tím nejkrásnějším bude posléze umění, jímž se budou knihy a jejich moudrost s podobnou rychlostí vtiskovat v lidskou duši, dopřeje-li Bůh, aby jednou bylo vynalezeno.*“ I když současný vynález toto Komenského předvídatelé přání nesplňuje, posunuje je možná dále k jeho uskutečnění.

Písmo, typografie, ilustrace a tisk mají ještě důležitého spojence, který umožňuje — vedle jiných faktorů — jejich reálný život. Je to materiál, na který tiskneme a ještě dlouho budeme tisk-

Obrázek 4: Odstavcová zarážka může mít různou podobu. Nejčastěji používané je odsazení prvního řádku dovnitř textu, ale někdy i vně, jako v tomto popisku. Též vertikální odsunutí odstavců je běžné. Méně časté je odsazení prvního řádku o šířku východového (posledního) řádku předchozího odstavce tak, jako v reprodukované ukázce z *Typographie* Oldřicha Hlavsy [12].

Též je možné oddělit odstavce speciálním znakem, nejčastěji ¶ (z latinského capitulum, anglicky zvaného pilcrow). Tento způsob se užíval hlavně ve středověku.

- Sloupce a buňky tabulky – mají zajistit jednoznačné oddělení sloupců a zarovnat hodnoty.
- Interpunkční mezerování – vychází z logického oddělení a tradice: např. tečka ukončuje předchozí větu, následující mezeru věty odděluje.
- Mezerování matematických výrazů – odlišně mezerujeme unární, binární a relační operátory. Mnoho práce tu vykoná  $\text{\TeX}$  sám, ale i tak zbývá dost ruční sazečské práce [31, 24], obr. 5.



Obrázek 5: Ukázka složitosti ruční matematické sazby s potřebnými výplňky. Už víte, proč se matematický text zapisuje v  $\text{\TeX}$ u mezi dolary? Ukázka z *Nauky o sazbě obyčejné, tabulkové, matematiky a chemie* Jindřicha Špalka z roku 1925 [23].

### 2.3. Jednotnost

Dalším principem dobrého dokumentu je *jednotnost*. Jednotnost se projevuje hned v několika rovinách. Každé se budeme věnovat zvlášť.

#### 2.3.1. Stejně elementy stejně

Formátování stejných elementů v celém dokumentu stejně je základní požadavek jednotnosti. Znamená to např. stejné odsazení odstavců, stejný způsob zvýraznění slov, stejné formátování titulků, stejnou sílu linek. Není-li dodrženo toto pravidlo, nebude struktura dokumentu jednoznačná a čtenář může být zmaten.

#### 2.3.2. Různé elementy se sjednocující myšlenkou

Dobrý typograf vtiskne dokumentu jednotnou myšlenku tím, že opakuje různé variace jednoho nápadu pro různé strukturální elementy. Přitom je třeba zřetelně zachovat hierarchickou strukturu dat. Příkladem může být zvýraznění titulku obrázku stejným ale menším písmem jako v nadpisu kapitoly.

Úprava dokumentu by měla být ve vzájemné harmonii. Zvláštní zřetel by měl být při tvorbě knih kladen na kompozici protilehlých dvoustran. Jednostranný tisk ztratil opodstatnění z důvodů ekonomických (cena papíru), ekologických (spotřeba

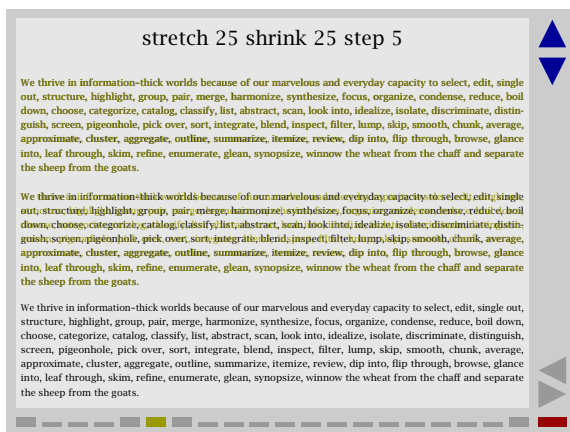
papíru), technických (zmizel průklep psacího stroje) i funkčních (přehlednost většího kusu textu bez listování). Obě stránky mají mít stejný počet řádek a pro umístění plovoucího objektu platí pravidlo, že má být vidět na dvoustraně, kde je odkazován.

### 2.3.3. Rovnoměrné pokrytí strany

Stránka při pohledu z větší vzdálenosti by měla působit stejnoměrnou šedou barvou, neměla by být roztráštěná. Taková působí chaoticky, jednotlivé strukturní elementy nejsou zřetelné. Naším cílem by měla být kompaktní strana. Prostředkem k jeho dosažení je citlivá práce s neviditelnými elementy.

Kontrast písma a pozadí by měl být co nejvyšší, ale na celé stránce podobný, jinak bude obojí nadměrně unavovat čtenářův zrak. Prostředkem je volba písma podobné síly, jak pro hlavní text, tak zvýraznění. Z tohoto důvodu se pro zvýraznění často používá italika stejného fontu; kresba je dostatečně odlišná, aby zvýraznění bylo jednoznačné, zároveň ale stejné optické hustoty. O prostrkávání Karel Dyrynk píše [5, str. 18]: *Vyznačování prostrkáním hodí se pro denní list nebo časopis, na jehož úpravě příliš nezáleží. Ale v klidné stránce knihy je takové roztrhané slovo světlé a rušivé, a zvlášť v krásné knize je prostrkání naprosto nepřijatelné.*

Kontrast textu též zvyšuje stará typografická finesa – sazba na *řádkový rejstřík*.



Obrázek 6: Jedním z předpokladů rovnoměrného pokrytí strany je neděravý odstavcový zlom. Při problematické sazbě (např. do úzkých odstavců) k němu napomáhá tzv. *hz-algoritmus* implementovaný do pdf $\TeX$ U [32, 27, 34, 22]. O optické vyrovnání hran textu se stará technika *prostrkání okrajů* [33]. Ukázka z testovacího dokumentu Hanse Hagena *Does pdf $\TeX$  make things better?* [7].

Při něm si řádky na rubu i líci stran přesně výškově odpovídají. Implicitní nastavení plain  $\text{T}_{\text{E}}\text{X}$ u i standardních tříd  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ u řádkový rejstřík nedodrží. Hlavním důvodem je problematická sazba matematických vzorců – původně hlavní účel  $\text{T}_{\text{E}}\text{X}$ u – do řádkového rejstříku. Zachovat řádkový rejstřík v  $\text{T}_{\text{E}}\text{X}$ u jde, ale ne triviálně, viz [19], [9,  $\text{C}_{\text{O}}\text{N}\text{T}_{\text{E}}\text{X}\text{T}$ ]. U víceloupcového textu je dodržení řádkového rejstříku estetickou nutností.

V publikacích, kde pracujeme s barvou (plakáty, časopisy, on-line dokumenty), je nutno věnovat pozornost barevnému kontrastu mezi textem a pozadím. Nápomocí mohou být programové pomůcky, např. webový Colour Contrast Visualiser [13].

Dalším prostředkem rovnoměrného pokrytí je neděravá sazba. Kromě velmi propracovaného odstavcového zlomu, který nám  $\text{T}_{\text{E}}\text{X}$  nabízí, můžeme využít mikrotypografické rozšíření pdf $\text{T}_{\text{E}}\text{X}$ u [28], viz obr. 6. Jinou možností, která byla hojně používána již prvním typografem Johannesem Gutenbergem, je použití písmenných variant, obr. 7. Jimi lze potlačit jak nevhodný zlom řádku, tak i tzv. řeky – několik mezerislovních mezer v odstavci nad sebou. Pokud je např. z důvodů velmi úzkého sloupce nemožné dosáhnout neděravého odstavce, je lepší místo sazby do bloku použít sazbu na praporek, viz popisek obr. 8.

#### 2.3.4. Dodržení národních, oborových a lokálních konvencí

Jednotnost by měla být zachována i v rámci vyšších celků, než je samotný dokument. Jde o národní typografické odlišnosti (pozor např. na velikost mezer okolo interpunkce, zejména pomlček [3]), dále o oborové konvence (značení bibliografie), o jednotné formátování publikací v edičních řadách (třeba diplomových prací [20]) a jednotlivých příspěvků ve sborníku. To vše splňuje očekávání čtenáře a napomáhá srozumitelnosti.

#### 2.3.5. Chronologická jednotnost

Dobrym rysem úpravy je chronologická korespondence mezi obsahem dokumentu a použitým písmem. Sázíme-li např. starý renezanční text, působí amatérsky, použijeme-li moderní písmo. Klasifikace písem je dobře popsána v [11, 2, 4].

### 2.4. Tradice, estetika

Měli bychom mít na paměti, že úprava dokumentů se vyvíjí po několik tisíciletí a za tu dobu přirozeně konvergovala podoba i umístění jednotlivých elementů k praktické použitelnosti. Designérské experimentování při formátování by mělo být prováděno s plným vědomím tohoto dlouhého vývoje.

Příkladem tradičního estetického hlediska je potlačení vdov (stránkový zlom za prvním řádkem odstavce...) a sirotků (... před posledním řádkem odstavce),

vě přikročeno k sazbě. Aby písmo bylo malebnější, nakreslil umělec od jednotlivých typů několik způsobů <e e e a a t t k k> a určil kde kterých upotřebovat, aby řádky byly světle vyrovnané, pokusmo stanovil šířku mezer mezi slovy, úpravu titulků atd. A pro každou

(a) O Preissigově úpravě *Slezských písní*.

Úmyslně zvolen větší stupeň, tercie, aby bylo dosaženo i vnějšího souhlasu s vážným, slavnostním, evangelickým obsahem knihy. Ježto by bylo nemožno tak velké a široké písmo typograficky správně sázet na šířku 20 cicer, bylo nutno je doplnit

e t, r ct st

(b) O Komenského *Kšaftu umírající matky jednoty bratrské* vydaného Spolkem českých bibliofilů roku 1909.

Obrázek 7: K vyrovnání světlosti řádek lze použít písmenné varianty. Některé nahrazují kerning, některé vyrovnávají řádku variantní šířkou. Ukázky z knihy Karla Dyrynka *Typograf o knihách* [5].

Domácí meta vznikne odstřížením rohů ze čtverce velikosti 43 cm, ostatní mety mají tvar čtverce o hraně 38 cm. Vnitřní hrana obdelníkového území pálkaře o rozměrech 91×213 cm je odsazena od hrany domácí mety o 15 cm a přečnívá o 91 cm roh hřiště.

Hrají jej dvě mužstva (oficiálně o devíti členech). Hráči družstva v poli (polaři) mohou být kdekoliv v poli, jen nadhazovač a chytač musí být ve svém území. Družstva se pravidelně střídají ve hře na pálce i v poli. Směna je období hry, kdy totéž mužstvo hrálo na pálce i v poli. Ke střídání úloh dojde, pokud družstvo v poli dosáhne třetího autu.

Účelem hráče na pálce je správně odpálit, proběhnout metovou dráhu (dotknout se v pořadí první, druhé třetí a domácí mety) dříve, než je vyaurován. Za takový oběh získá jeho mužstvo bod. Úkolem polařů je zabránit běžci v získání bodu tím, že jej autují dříve, než dokončí oběh.

#### Nadhazování

● Nadhazovač musí zaujmut postavení oběma nohama na zemi a musí se nohama dotýkat nadhazovací mety.

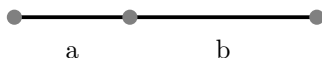
Obrázek 8: Zcela kompaktním stranám napomáhali hlavně středověcí typografové výplní odstavcových zarážek, čímž dosahovali obdelného tvaru odstavců. Tento přístup byl použit pro sazbu drobné brožurky Pravidla softballu [6], jejíž stránku zde reprodukuje. Využívá se ‚pružného‘ obrázku pomocí `\cleaders`:

```
\def\softbat{\batA\nobreak
\cleaders\hbox{\batB}\hfil\nobreak%
\batC\nobreak
\cleaders\hbox{\batD}\hskip0pt plus1.5fil
\nobreak\batE}
```

kde `\bat?` jsou jednotlivé dílky pálky.

obecně parchantů. Sázíme-li do řádkového rejstříku, můžeme změnit délku textu změnou zlomu dlouhého odstavce pomocí `\looseness`, přidat půlřádek okolo nadpisu kapitoly nebo upravit velikost obrázku.

Těžko formulovat nějaká obecná pravidla tohoto principu. Charakterizuje jej střídmost, dodržování vzájemných proporcí, kompozice (viz obr. 9) a celková harmonie. Pro náš návrh se můžeme inspirovat bezpočtem dobrých i špatných příkladů.



Obrázek 9: Při komponování se často využívá čistých geometrických tvarů a někdy i hudební harmonie [2]. Jednoduchým příkladem je formátování nadpisů na střed. Estetické vnímání však zcela nekopíruje naprostou symetrii. Tak jako lépe vnímáme, když je fotografovaný objekt vychýlený z geometrického středu, tak třeba i horizontální umístění stránkové číslice nebo vertikální poloha marginálie je lepší mimo střed či kraj. Nápomocí nám může být tzv. *zlatý řez*. Pro něj platí formule  $b : a = (a + b) : b$  a poměr stran  $a : b \approx 0,618$ . Šířka obr. 2 je komponována právě jím.

## 2.5. Obecné pravidlo formátování

Při pozorném přečtení všech čtyř principů formátování zjistíme, že jednotlivá pravidla si vzájemně protirečí. Zcela jednotně pokrytá stránka nepopisuje strukturu dat a dotaženo ad absurdum, naprosto homogenně šedá stránka je sice krásně jednotná, ale nepřináší (téměř) žádnou informaci. Pro aplikování pravidel je proto třeba si uvědomit, že jejich platnost je hierarchická; přednost je v tom pořadí, v jakém jsou zde prezentovány. Koliduje-li tedy princip informace a struktura (což nebývá moc časté), je třeba dát přednost informaci. Pokud je v rozporu vyjádření struktury a jednotnosti, je třeba dát přednost struktuře ap., obr. 10. Protože tuto kolizi musí sazeč řešit každodenně, shrneme si ji následujícím tvrzením: *Povolme právě tolik výjimek z uniformity, kolik je nezbytné pro vyznačení zvolené struktury; ne více.* Proto je žádoucí použít právě jednu odstavcovou zarážku z těch zmíněných v popisku obr. 4.

Závěrem této kapitoly poznamenejme, že výše popsaná pravidla jsou pouze zobecnujícím návodem, nikoliv dogmatem. Úprava dokumentů patří mezi umělecká řemesla a neplatí proto pro ně striktní pravidla. Formalizovali jsme je, abychom z nich vyvodili následující požadavky na editaci dat.

Výška znaku integrálu  $\int$  ve vzorcích na zvláštním řádku je 26 bodů bez ohledu na výšku vzorce; v textu, kde se nemají vyskytovat zlomky s čitatelem a jmenovatelem nad sebou, nýbrž (v nezbytných případech) se šikmým lomítkem nebo se záporným exponentem, je znaménko integrálu  $\int$  12bodové, a to i v případě, že má meze.

$$\int_{-1}^1 \left( \frac{\partial^2 u}{\partial x \partial t} \right)^2 dx ; \int_{-1}^1 (\partial^2 u / \partial x \partial t)^2 dx .$$

Obrázek 10: Tento příklad ilustruje rozpor mezi principy informace a jednotnosti. Velký matematický symbol by při zachování jednotného řádkování způsobil slítí s textem okolních řádek, nečitelnost a proto ztrátu informace. V takovém případě je třeba oželet pravidelné řádkování. Mnohem lepším řešením je nahradit tento symbol menší variantou či vložit jej na zvláštní řádek (display mód). Ukázka z knihy Karla Wicka *Pravidla matematické sazby* [31].

### 3. Požadavky na editaci dat

Naplnění výše popsaných principů není jednoduché vzhledem k jejich počtu, variabilitě, vzájemnému provázání a dělbě kompetencí mezi několik osob (autoři, redaktor, sazeč, korektor). Práci by usnadnily nástroje, které navádějí autory k přípravě strukturovaných zdrojových textů dokumentů (dále rukopisů) a systémově zjednodušují práci sazečům. Pokusme se požadavky na takové nástroje a formáty dokumentů formulovat. Učiníme tak zvlášť pro autory dokumentů a zvlášť pro sazeče a správce dat.

#### 3.1. Z hlediska autorů dokumentů

Většinou nelze předpokládat, že autor je zároveň IT specialista. Proto nástroj pro tvorbu rukopisů musí být uživatelsky co nejjednodušší a intuitivní. Domnívám se, že jeho ovládání musí vycházet z všeobecně propagované počítačové gramotnosti ovládání toho pravého a jediného textového procesoru.

Zároveň však musí autora nabádat k *výhradnímu používání strukturních elementů*. Vkládání grafických elementů by mělo být značně ztíženo nebo dokonce zakázáno. Tento požadavek je zcela v protikladu s praxí současných textových WYSIWYG procesorů.

Takto důsledně vnucená logická struktura dokumentu přináší dvě pozitiva: a) systémově umožní jednotný grafický vzhled stejných elementů a b) oddělí a odloží formátování dokumentu do dalšího kroku. Separace funkcí podporuje celkovou kvalitu dokumentu, protože o každou složku se postará specialista: o obsah autor a o formátování typograf, sazeč či designér. Hranice kompetencí je přitom jasně definovaná.

Nástroj by též měl asistovat ve vkládání náležitých elementů (tj. povolit vložit jen ty dle kontextu povolené), jejich automatické číslování (stránky, odrážky), generování (datum, obsah, křížové odkazy, citace), zpracování (řazení) a verifikaci (struktura, url). Vše centralizované do jednoho místa co nejlépe relevantní obsahové informaci.

Požadavky z hlediska autorů shrneme do těchto bodů:

- snadnost a intuitivnost použití,
- výhradní používání strukturních elementů s jejich kontextovou nabídkou,
- zákaz či potlačení grafických elementů,
- hlídání udržování definované struktury textu,
- odstranění rutinních a k chybám náchylných úkonů (číslování, obsah).

### 3.2. Z hlediska sazeče a správce dat

Úkolem sazeče je vtisknout dokumentu pěknou a jednotnou tvář. Mezi požadavky proto patří možnost využít všechny dostupné grafické elementy. Technika by neměla omezovat návrháře.

Aby byla zaručena jednotnost formátování, neměl by být graficky upravován každý strukturní element zvlášť, ale centrálně pro všechny instance jednoho strukturního elementu. Formátovací práce tak vyžaduje vyšší míru abstrakce, ale kromě vyšší efektivity u rozsáhlých dokumentů je vyvážena možností experimentovat s formátovacími variantami na celém dokumentu najednou.

Protože struktura dokumentu a jeho formátování se vzájemně ovlivňují, je v praxi výhodné, když sazeč může experimentovat s úpravou již od počátku tvorby dokumentu a flexibilně navrhovat autorům vhodné strukturní elementy. I zde je proto nezbytné, aby formátování bylo oddělené od dat, aby autoři úpravou textu nezrušili všechnu sazečem dosud vykonanou práci. Požadavky sazeče můžeme shrnout do těchto bodů:

- grafická obecnost,
- podpora jednotnosti,
- snadnost experimentování,
- oddělení formátování od obsahu,
- čitelnost a
- minimalizace práce.

Kromě toho musíme připočítat požadavky správce dat. Jsou jimi

- dlouhodobá platnost dat (využívání standardů),
- multiúčelovost (zpracování variabilním softwarem, různý výstup, otevřený formát),
- validace definované struktury dat a
- platformní nezávislost.



## 4. Strukturní značkování

Jak záhy uvidíme, strukturní značkování je způsob vytváření dokumentu, který dobře vyhovuje výše uvedeným požadavkům. Spočívá v doplnění textu o předem stanovené značky vyznačující jednotlivé strukturní elementy.

### 4.1. Výhody a nevýhody strukturního značkování

Strukturně označovaný dokument přináší tyto výhody:

- oddělení formátování od obsahu,
- oddělení formátování od struktury,
- expertní rozdělení kompetencí pro autora a sazeče,
- snadné udržování jednotnosti,
- vyšší přehlednost u větších projektů a pro více uživatelů,
- definování a verifikování struktury a platnosti dat,
- možnost globální změny formátování,
- snadné generování různých výstupů z jednoho zdroje (znovuvyužití textu),
- zvýšení efektivity psaní využíváním předdefinovaných elementů,
- snazší správa (jazykových) variant,
- vyšší časová stálost (snazší archivace).

A tyto nevýhody:

- požaduje větší disciplínu autora,
- menší flexibilitu u elementů, na které se nemyslelo dopředu.

Obecně lze říci, že výhody vedou k vyšší efektivitě zpracování dokumentu, ačkoliv to vyžaduje více počátečního úsilí.

### 4.2. Formáty

Obecným standardním formátem strukturního značkování je SGML (Standard Generalized Markup Language), ale největší popularitu získaly jeho zjednodušené aplikace HTML (HyperText Markup Language) a XML (Extensible Markup Language) [14]. Protože HTML míchá strukturní a grafické značky, je XML dnes de facto standardem strukturního značkování (obr. 11).

Pro značkování jazykem XML je nejprve třeba definovat hierarchickou strukturu dokumentu jazykem DTD (Document Type Definition).<sup>3</sup> Naše datové XML pak bude s tímto souborem nerozlučně spjata. Pokud najdeme pro náš typ dokumentu definici s vhodnými elementy, můžeme ji použít a nemusíme si ji vymýšlet. Příkladem je běžně používaná definice DocBook [18, 30] zvláště vhodná pro odborné texty a manuály. Pro jednoduché či netypické dokumenty lze však DocBook přirovnat ke střílení kanónem na komára.

---

<sup>3</sup>Jinou možností je XML Schema nebo Relax NG.

```

<!-------Uvod-----> Uvod -->
<chapter>
<title>Uvod</title>
<p>
Cílem projektu <em>Analýza meteorologických
analýg</em> (zkratkou: <em>ANALOG</em>; anglického názvu <em>Meteorologic
ANALogy Analysis</em>) byla odvozena zkratka MANA-ANA. </Footnote>
bylo vytvoření nástroje pro vyhledávání „podobných“ meteorologických
situací v databázi meteorologických měření
<a href="http://www.cdc.noaa.gov/cdc/data.ncep.reanalysis.html"
>NCEP/NCAR</a>
amerického střediska
<a href="http://www.cdc.noaa.gov/sc1imate_diagnostics_center">
Podobnost meteorologických situací je odvozena od použité podobnostní
funkce. V systému jsou implementovány dvě základní podobnostní funkce:
RMS rozdíl dvou měření a RMS úhlu mezi normálními daty měření.
</p>
<p>
Systém dále umožňuje:
<itemize>
<li>Automatické stahování (aktualizací) dat z NCEP/NCAR
databáze. </li>
<li>Volbu meteorologických veličin, referenčního období,
časového rozsahu prohlédávání, geografické oblasti a typu
podobnostní funkce. </li>
<li>Vyhodnocení podobnostní funkce a následně seřazení situací
na základě podobnosti. </li>
<li>Vizuální zobrazení výsledků. </li>
</itemize>
</p>
<p>
Projekt byl navržen jako serverová webová aplikace. Tato realizace
umožňuje uživatelům vzdálený přístup prostřednictvím webového rozhraní
</p>
UTF8-----L49 C20 3% manaana.xml (...:nule/meteo/src/doc) (XML [doc] F

```

(a) XML

```

<ENTITY fn "fn" >
<ENTITY sum "sum" >
<ENTITY yes "yes" >
<ENTITY no "no" >
<ENTITY lang "Cz|en|un" >
<ENTITY lang-default "un" >
<ENTITY x-item-type "0|1" >
<ENTITY x-item-type-default "0" >
<ELEMENT doc
(title, subtitle?, authors, copyright?, baseName?, dir-fig?, body)>
<ATTLIST doc version CDATA #REQUIRED>
<ATTLIST doc charset CDATA #IMPLIED>
<ELEMENT OUT-HTML (#PCDATA)>
<ELEMENT OUT-PDF (#PCDATA)>
<ELEMENT titled (#PCDATA)>
<ELEMENT titled_label
(ATTLIST titled_label CDATA #IMPLIED)
<ELEMENT subtitle (#PCDATA)>
<ELEMENT authors (Institution?, Logo?, author->)>
<ELEMENT dir-fig (#PCDATA)>
<ELEMENT institution (#PCDATA)>
<ELEMENT logo (#PCDATA)>
<ELEMENT author (name, email?, address?)>
<ELEMENT name (#PCDATA)>
<ELEMENT email (#PCDATA)>
<ELEMENT address (#PCDATA)>
MSW-MB: T="--L27 C0 20% manaana.dtd (...:text-cstug08/figs) (DTD Font

```

(b) DTD

```

<hr />
<h2>1. Uvod</h2>
<p>
Cílem projektu <em>Analýza meteorologických analýg</em>
<em>ANALOG</em> (zkratkou: <em>ANALOG</em>; anglického názvu <em>Meteorologic
ANALogy Analysis</em>) byla odvozena zkratka MANA-ANA.
<a href="http://www.cdc.noaa.gov/cdc/data.ncep.reanalysis.html"
>NCEP/NCAR</a>
amerického střediska
<a href="http://www.cdc.noaa.gov/sc1imate_diagnostics_center">
Podobnost meteorologických situací je odvozena od použité
podobnostní funkce. V systému jsou implementovány dvě
základní podobnostní funkce: RMS rozdíl dvou měření a RMS
úhlu mezi normálními daty měření.
</p>
<p>
Systém dále umožňuje:
<ol style="list-style-type: disc" type="0">
<li>Automatické stahování (aktualizací) dat z NCEP/NCAR
databáze. </li>
<li>Volbu meteorologických veličin, referenčního období,
časového rozsahu prohlédávání, geografické oblasti a typu
podobnostní funkce. </li>
<li>Vyhodnocení podobnostní funkce a následně seřazení
situací na základě podobnosti. </li>
<li>Vizuální zobrazení výsledků. </li>
</ol>
</p>
<p>
Projekt byl navržen jako serverová webová aplikace. Tato
realizace umožňuje uživatelům vzdálený přístup
prostřednictvím webového rozhraní bez závislosti na
konkrétním operačním systému. Uživatelské rozhraní
je dvojjazyčné, české a anglické.
</p>
MIME/Ltn-2-----L38 C28 2% manaana-doc.html (...:cstug08/figs) (HTML

```

(c) HTML

```

%-----Uvod-----
\chapter{Uvod}
Cílem projektu <em>Analýza meteorologických
analýg</em> (zkratkou: <em>ANALOG</em>; anglického názvu <em>Meteorologic
ANALogy Analysis</em>) byla odvozena zkratka MANA-ANA.
bylo vytvoření nástroje pro vyhledávání „podobných“ meteorologických
situací v databázi meteorologických měření
\href{http://www.cdc.noaa.gov/cdc/data.ncep.reanalysis.html}
{NCEP/NCAR} amerického střediska
\href{http://www.cdc.noaa.gov/sc1imate_diagnostics_center}
{Climate Diagnostics Center}.
Podobnost meteorologických situací je odvozena od použité podobnostní
funkce. V systému jsou implementovány dvě základní podobnostní funkce:
RMS rozdíl dvou měření a RMS úhlu mezi normálními daty měření.
Systém dále umožňuje:
\begin{itemize}
\item Automatické stahování (aktualizací) dat z NCEP/NCAR
databáze.
\item Volbu meteorologických veličin, referenčního období,
časového rozsahu prohlédávání, geografické oblasti a typu
podobnostní funkce.
\item Vyhodnocení podobnostní funkce a následně seřazení situací
na základě podobnosti.
\end{itemize}
Projekt byl navržen jako serverová webová aplikace. Tato realizace
umožňuje uživatelům vzdálený přístup prostřednictvím webového rozhraní
bez závislosti na konkrétním operačním systému. Uživatelské rozhraní
je dvojjazyčné, české a anglické.
%-----Spuštění programu-----
\section{Spuštění programu}
UTF8-----L18 Cl3 Top ex-latex.tex (...:text-cstug08/figs) (TeX Font F1

```

(d) L<sup>A</sup>T<sub>E</sub>X

Obrázek 11: Příklady strukturně označovaného textu a definice jeho struktury.

XML je bezkompromisní strukturně značovací jazyk, netvrdím však, že je vždy nutné psát dokument v něm. Pokud jsme sami sobě autory i sazeči drobného článku psaného T<sub>E</sub>Xem, byla by užvaněnost XML spíše překážkou. Poměrně dobře lze strukturně značkovat i v T<sub>E</sub>Xu. Vystačíme si se stylovým souborem, kde si makrojazykem T<sub>E</sub>Xu definujeme všechny potřebné a implicitně nenabízené strukturní značky (v plain T<sub>E</sub>Xu nám to dá více úsilí, protože implicitně téměř žádné takové značky nenabízí); tento soubor pak vložíme pomocí \input do vlastního zdrojového souboru se strukturně označovaným textem. Dbáme, abychom

v něm nepoužili žádný formátovací příkaz a centralizovali příbuznou informaci do jednoho místa, viz obr. 12.<sup>4</sup> Např. namísto `\vspace` píšeme `\bigskip` nebo `\includegraphics[width=5.47cm]{ma/strasna/cesta/soubor.pdf}` nahradíme definicí

```
\graphicspath{{ma/strasna/cesta/}}
\includegraphics[width=.5\linewidth]{soubor}
nebo ještě lépe
\newdimen\FigSize \FigSize=.5\linewidth
\includegraphics[width=\FigSize]{soubor}
```

Nevýhodou tohoto přístupu je nízká formalizace a tudíž neexistence přímého validačního nástroje.

Strukturální značkování již začaly nabízet i textové procesory, ale použitelnost je mizerná. Uživatel musí projevit nebývalou disciplínu a znalost, aby nesklouzl k použití – v okenní liště snadno přístupných – formátovacích příkazů (obr. 13). I zpracovatel takto označovaných dat není v lehké pozici, neboť formát dat je dosti kryptický, ve své obecnosti značně složitý a často obsahuje „smetí“ vzniklé editací (např. v podobě prázdných elementů).

<sup>4</sup>V praxi zjistíme, že v závěrečné fázi formátování potřebujeme učinit drobné výjimky z obecného formátování, abychom vyhověli typografickým požadavkům. Třeba vložit `\looseness` pro zamezení vzniku parchanta. Vyplatí se i tyto příkazy strukturně označovat speciální značkou, např. `\finalTYP0a{\looseness=-1\par}` při definici `\long\def\finalTYP0a#1{#1}`. Jedna taková značka odpovídá jednomu způsobu formátování. V případě potřeby se pak dají tyto formátovací výjimky přepínat/zneaktivnit předefinováním `\long\def\finalTYP0a#1{}`.

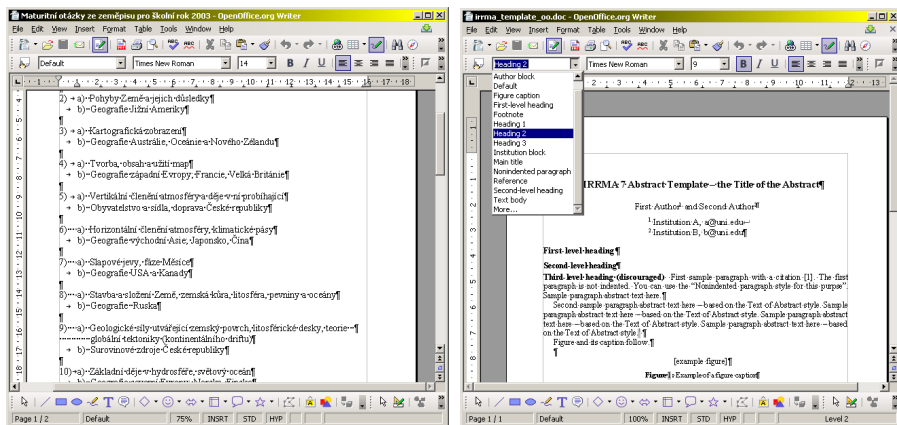
```
% Oblasť (První pomoc){pomoc}
%bod{Ošetření lehkých poranění}{
%podminky{Umí ošetřit oděry, povrchovou ránu, opruzení, puchýř a kŮ
%z-nosu. Dovede odstranit třísku, klíště; provést dezinfekci
%nepoškození tetanu.}
%korekvizita{!Poranění}
%znacka{!Poranění}
%bod{Základní životní funkce}{
%podminky{Zná základní životní funkce. Dovede provést umělé dýchání
%zevní masáž srdce a zastavit silné krvácení.}
%korekvizita{!Poranění}
%znacka{!Zivotnifunkce}
%bod{Ošetření nohou po pochodech}{
%podminky{Umí vybrat vhodnou obuv-nápravu a ošetřit nohy před
%pochodem i po něm. Prevence a ošetření puchýřů.}
%}
%bod{Jednoduché obvazy}{
%podminky{Umí prakticky použít šátkové obvazy.}
%korekvizita{!Obvazy}
%znacka{!Obvazy}
%bod{Nejběžnější léky}{
%podminky{Zná použití a dávkování nejběžnějších léků (acylglyrín,
%živčištěné uhlí, borová voda, dezinfekční prostředky na kŮ
%korekvizita{!Lekarna}
%znacka{!Lekarna}
%}
%bod{Nejběžnější léky}{
%podminky{Zná použití a dávkování nejběžnějších léků (paralen,
%živčištěné uhlí, ophal, dezinfekční prostředky na rány.}
%korekvizita{!Lekarna}
%znacka{!Lekarna}
%}
MIME/Ltn-2T-----L445 c0 27% stz08.tex (C:\tex\ap1\kace\stzeka) CTeX
```

(a) Datový soubor

```
\def\korekvizita#1{\expandafter\korekvizitb #1,\endprekeli}
\def\korekvizitb#1,#2,\endprekeli{
%globa\setbox\korekbox\hbox{\unhbox\korekbox%
%expandafter\FX\csname #1\endcsname\velex
%message\space korekvizita #1 nedefinovana!}
%else{hskiptsty\korekvizitn %
%$csname #1\endcsname}
%$csname #1\endcsname{}}\fi}%
\setbox\pombox\hbox{#2}
\ifdim\wd\pombox>0pt,\let\next=\korekvizitc
%else,\let\next=\korekvizitd\fi\next{#2}}
\def\korekvizitc#1{\expandafter\korekvizitb #1,\endprekeli}
\def\korekvizitd#1{\global\setbox\poznamkabox\hbox{{\poznamfn #2}}}
\def\podminkyparagraf{base1nestip=2ex \parindent=-1em\ovalpresah
%nedel \parsetip=0.8base1nestip plusipr mfnustip}
\def\podminky#1{\global\setbox\podminkybox\hbox{{\let\F=\\podminkyntu
%def\podminkyntu{\ifdim\wd\podminkybox>0pt
%globa\setbox\podminkybox\hbox{\egroup\podminkyparagraf
%unhbox\podminkybox}\podminkyntu\hskipt=4em%
%$box\tolem{WF1}\thebox\hskipt=0.8em%
%unhbox\podminkybox\par}\egroup}\fi}}
\let\novastrasna\veject
\def\podpisfbox#1{\wbox\to#1\wss\vf11}
\hrule\wbox\to\podpisize{\vrule\wbox\to2.4ex\vf11}\hrule\w
\vf11\hrule\vf11}}
\def\bodye2{
%globa\inputput={\inputput}
%if\defe1\adnanca\bodye2\bv 1 #?%
MIME/Ltn-2-----L546 c0 66% stzeka.tex (C:\tex\ap1\kace\stzeka) CTeX
```

(b) Definiční soubor

Obrázek 12: Příklad důsledného značkování v plain  $\TeX$ U. Poměrně mnoho práce s vytvořením definičního souboru je kompenzováno faktem, že datový soubor může snadno editovat kdokoliv i bez znalosti  $\TeX$ U.



(a) Špatné

(b) Dobré

Obrázek 13: Příklady špatného a dobrého značkování v OpenOffice.org.

```

BOOK{ Dyrnyk93,
  AUTHOR = { karel dyrnyk },
  TITLE = { Typograf a knih(\a)ch },
  PUBLISHER = { Kletaur Polygrafie, Praha },
  YEAR = { 1993 },
  EDITION = { 2. },
  KEYWORDS = { sazba, typografie, uprava knih },
  NOTE = { Prvn(\i) vyd(\a)n(\i) 1911. },
  WHO = { Zyka },
}

BOOK{ hlavsaz6,
  AUTHOR = { old(v r)ich hlavsaz and karel wick },
  TITLE = { Typografia 2 },
  SUBTITLE = { P(\i)jso, ilustace, kniha },
  PUBLISHER = { Nakladatelstv(\i) technick(\e) literatury, Praha JU },
  YEAR = { 1976 },
  EDITION = { 1. },
  KEYWORDS = { sazba, typografie },
  WHO = { Zyka },
}

BOOK{ wick66,
  AUTHOR = { karel wick },
  TITLE = { Pravidla matematick(\e) sazby },
  PUBLISHER = { Academia, Praha },
  YEAR = { 1966 },
  EDITION = { 1. },
  KEYWORDS = { sazba, typografie, matematika },
  WHO = { Zyka },
}

MANUAL{ stary05,
  AUTHOR = { richard star(\v) }.
}
MSW-MB-----L137 C0 16% tex.b1b (z:\b1b\ex\b1b) (e1b\ex Font)-----

```

(a) BIB

Obrázek 14: I starý známý BibT<sub>E</sub>Xový formát je strukturně značkový text.

### 4.3. Prostředky

Zde uvedený seznam prostředků pro editaci, zpracování a formátování strukturně značkových dat není a ani nemůže být úplný (obr. 14). Slouží pro nasměrování čtenáře. Největším problémem se autorovi jeví snadno dostupný strukturní open XML editor, s ovládnáním nepříliš vzdáleným textovým procesorům.

### 4.3.1. Editory

Pro vytváření strukturně značkových dokumentů lze používat různé typy editorů. Pokusíme se je stručně charakterizovat.

- a) Klasický textový:
  - + lze zapsat všechny možnosti formátu (je-li tento textový),
  - zcela nutná znalost formátu,
  - ukecanost formátu může vést až k praktické nepoužitelnosti,
  - bez přímé validace,
  - bez grafického formátování.
- b) Textový s podporou pro strukturní formát (nabídka, klávesové zkratky) [např. Emacs, Vim, jEdit]:
  - + bez omezení na možnosti formátu,
  - + kontextová nabídka,
  - + strukturní pohyb,
  - + syntaktické zvýraznění,
  - + validace,
  - nutná znalost formátu,
  - bez grafického formátování.
- c) Klasický WYSIWYG editor [Microsoft Word, OpenOffice.org Writer, LyX]:
  - + snadná editace,
  - bez přímé validace,
  - nenabádá ke strukturnímu značkování,
  - problematické získání strukturovaného obsahu.
- d) Strukturní WYSIWY(G|M) editor [Syntext Serna, oXygen, Altova, XML-mind]:
  - + vstup jen dovozené struktury (on-line validace),
  - + kontextová nabídka,
  - + strukturní pohyb,
  - + není nutná znalost formátu,
  - + grafické formátování,
  - z formátu použitelné, jen co je implementováno.

Vytvářet dokumenty pomocí editorů typu b) může být velice efektivní, je třeba se jen naučit několik nejpoužívanějších klávesových zkratk. Práce s tímto typem editorů však není v osnovách výuky počítačové gramotnosti a těžko si například představit, že bychom žádali každého autora sborníkového příspěvku, aby jej použil s dodaným DTD. Ten většinou není schopen použít ani dodanou šablonu do textového procesoru typu c).

Nadějná je skupina d). V nich na pozadí editace probíhá transformace XML → XML FO (Format Object) pomocí ručně či grafickým nástrojem připraveného XSL (eXtensible Stylesheet Language). Specifikuje se v něm, jak má obsah toho kterého elementu při editaci vypadat. Např. že `<chapter-title>` má být

na samostatném vertikálně odděleném řádku zobrazen větším tučným písmem. Uživatel běžně nemá ke změně tohoto písma přístup, může pouze volit z kontextově dostupných strukturních elementů. Grafické formátování přibližuje editaci textovému procesoru, čímž je pro běžného uživatele stravitelnější. Výstupním formátem je – po odstranění obalových FO – validní XML.

#### 4.3.2. Zpracování

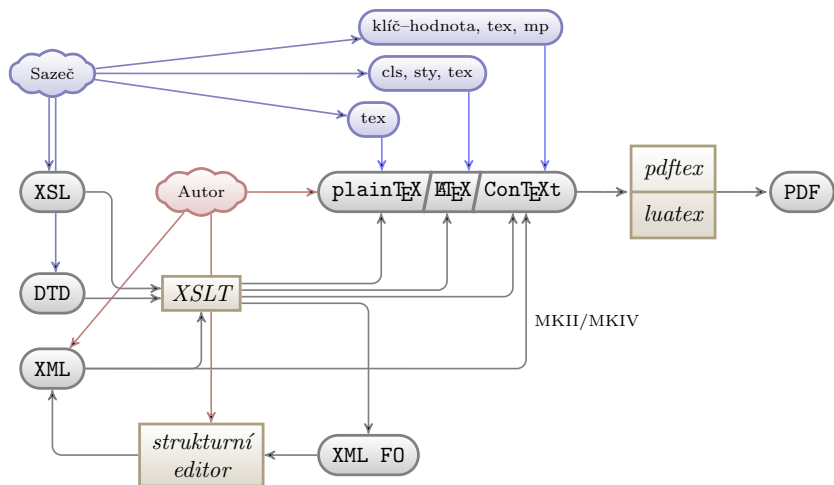
Efektivní a značně flexibilní nástroj pro konverzi a zpracování XML dokumentu je XSLT (eXtensible Stylesheet Language Transformations) procesor, např. Saxon. Snadno jím můžeme transformovat náš zdrojový text do  $\text{T}_{\text{E}}\text{X}$ u/ $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ u/ $\text{C}\text{O}\text{N}\text{T}_{\text{E}}\text{X}\text{T}$ u, do (X)HTML, nebo provést výběr či řazení dat. Některé zpracování XML souboru zvládá  $\text{T}_{\text{E}}\text{X}$  sám. Asi největší možnosti nabízí nová verze  $\text{C}\text{O}\text{N}\text{T}_{\text{E}}\text{X}\text{T}$ u MKIV založená na  $\text{L}\text{u}\text{a}\text{T}_{\text{E}}\text{X}$ u [10].

Protože většina strukturně značkovacích formátů je textová, na mnohé zpracování těchto dokumentů můžeme snadno využít skriptovací jazyky (jako Perl, AWK, Ruby, Lua, ...).  $\text{T}_{\text{E}}\text{X}$ istům jsou dobře známé nástroje Bib $\text{T}_{\text{E}}\text{X}$  a Make-Index, načítající strukturovaná data vytvářená samotným  $\text{T}_{\text{E}}\text{X}$ em (obr. 14). Na tomto příkladě je vidět, že mnoho práce vykoná  $\text{T}_{\text{E}}\text{X}$  sám, nicméně pro mnohé úlohy, např. řazení, je efektivní využít externí nástroj a několik průchodů  $\text{T}_{\text{E}}\text{X}$ u. S příchodem  $\text{L}\text{u}\text{a}\text{T}_{\text{E}}\text{X}$ u se toto opakované startování  $\text{T}_{\text{E}}\text{X}$ u a externích programů může odstranit. Ubyde dočasných souborů, závislosti na jiných programech a celá kompilace se zrychlí.

#### 4.3.3. Formátování

Existuje mnoho způsobů jak formátovat strukturně značkováný dokument  $\text{T}_{\text{E}}\text{X}$ em. O některých obecně pojednávají články [17, 15, 16, plain,  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ ], [8, 10,  $\text{C}\text{O}\text{N}\text{T}_{\text{E}}\text{X}\text{T}$ ], prakticky pak [29].

Na obr. 15 je ukázáno několik přístupů, které autor tohoto textu používá nebo je považuje za perspektivní. Dobře vidíme dělbu práce mezi autorem a sazečem i úroveň znalostí, které musí oba znát k zvládnutí své práce. Např. sazeč formátující v plain  $\text{T}_{\text{E}}\text{X}$ u musí znát makrojazyk  $\text{T}_{\text{E}}\text{X}$ u. Autor je plně závislý na míře abstrakce, kterou mu sazeč svými definicemi připraví. V  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ u si autor vystačí s volbou třídy, balíčků a základní definicí zkratk; sazeč však kromě makrojazyka  $\text{T}_{\text{E}}\text{X}$ u musí zvládnout upravovat (složité) makra jádra  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ u. Autor  $\text{C}\text{O}\text{N}\text{T}_{\text{E}}\text{X}\text{T}$ ového textu je na tom podobně jako autor využívající  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ , ale sazeč je ve výhodě. Spoustu designérské práce udělá pomocí uživatelské parametrizace klíč-hodnota bez znalosti jazyka  $\text{T}_{\text{E}}\text{X}$ u. Grafickou složku návrhu realizuje v jazyce  $\text{M}\text{E}\text{T}\text{A}\text{P}\text{O}\text{S}\text{T}$ . Pro méně běžnou sazbu je však i zde netriviální znalost  $\text{T}_{\text{E}}\text{X}$ u nutností.



Obrázek 15: Datový tok přípravy a formátování strukturně značkováného dokumentu  $\TeX$ em. Znázorněna je dělba práce mezi autorem a sazečem. Profese redaktora, designéra a korektora jsme záměrně vypustili ze dvou důvodů: za prvé většinou komunikují jinými prostředky než přímou editací dokumentu, na kterou se zde zaměřujeme, a za druhé pro zachování únosné složitosti diagramu.

## 5. Závěr

V článku jsme ukázali, jak je užitečné oddělit obsah dokumentu od formátování použitím strukturního značkování. Musíme však konstatovat, že nevíme o žádném nekomerčním editoru, který by byl vhodný pro autora odmítajícího své dokumenty ručně značkovat. Do doby, než se objeví a realizuje open free projekt WYSIWYG strukturního editoru, zůstane strukturní značkování doménou IT specialistů nebo velkých koncernů a státních institucí, kteří využívají pro přípravu dokumentů komerční řešení.

## Reference

- [1] Bohuslav Blažej. *Grafická úprava tiskovin*. SPN, 1. edition, 1990.
- [2] Robert Bringhurst. *The Elements of Typographic Style*. Hartley & Marks, Point Roberts, WA, USA, version 2.4 edition, 2001.
- [3] Karel Dostál. O jedné vraždě a jejím možném odčinění. *Zpravodaj Česko-slovenského sdružení uživatelů  $\TeX$ u*, 4(4):163–173, 1994.
- [4] Jean-Luc Dusong and Fabienne Siegartová. *Typografie. Od olova k počítačům*. Svojtka a Vašut, 1. edition, 1997.

- [5] Karel Dyrnk. *Typograf o knihách*. Kentaur Polygrafia, Praha, 3. edition, 1993. První vydání 1911.
- [6] Jiří Gerlich and Vít Zýka. Pravidla softballu, March 1998. <http://zyka.net/soft/soft-avan.pdf>
- [7] Hans Hagen. Does pdf $\TeX$  make things better?, 1999. <http://www.pragma-ade.com/pdftex/pre-hz.pdf>
- [8] Hans Hagen. *XML in Con $\TeX$ t*, 2001. <http://www.pragma-ade.com/general/manuals/example.pdf>
- [9] Hans Hagen. *It's in the details*, 2003. <http://www.pragma-ade.com/general/manuals/details.pdf>
- [10] Hans Hagen. *Dealing with XML in Con $\TeX$ t MKIV*, 2008.
- [11] Oldřich Hlavsa and František Sedláček. *Typografická pisma latinková*. Státní nakladatelství technické literatury, Praha, 1. edition, 1957.
- [12] Oldřich Hlavsa and Karel Wick. *Typographia I*. Nakladatelství technické literatury, Praha, 1. edition, 1976.
- [13] Thomas Hooper. Colour contrast visualiser. <http://www.stainlessvision.com/files/ColourContrastVisualiser.swf>
- [14] Jiří Kosek. Lehký úvod do XML. *Zpravodaj Československého sdružení uživatelů  $\TeX$ u*, 10(4):122–132, 2000.
- [15] Jiří Kosek. Jade $\TeX$ . *Zpravodaj Československého sdružení uživatelů  $\TeX$ u*, 13(1):15–26, 2003.
- [16] Jiří Kosek. Passive $\TeX$ . *Zpravodaj Československého sdružení uživatelů  $\TeX$ u*, 13(1):26–38, 2003.
- [17] Jiří Kosek. Použití parseru XML v  $\TeX$ u. *Zpravodaj Československého sdružení uživatelů  $\TeX$ u*, 13(1):6–14, 2003.
- [18] Jiří Kosek. *DocBook: Stručný úvod do tvorby a zpracování dokumentů*, 2007. <http://www.kosek.cz/xml/db/>
- [19] Petr Olšák.  *$\TeX$ book naruby*. Konvoj Brno, 1. edition, 1997.
- [20] Eduard Polách. *Pravidla sazby diplomových prací*, 2000. <http://home.pf.jcu.cz/~edpo/pravidla/pravidla.pdf>
- [21] Pavel Pop, Jindřich Fléger, and Vladimír Pop. *Ruční sazba I*. Státní pedagogické nakladatelství Praha, 2. edition, 1989.
- [22] R Schlicht. *The microtype package*, 2008. <http://www.ctan.org/get/macros/latex/contrib/microtype/microtype.pdf>.
- [23] Jindřich Špalek. *Nauka o sazbě obyčejné, tabulkové, matematiky a chemie*. Spolek faktorů knihtiskáren, Praha, 3. edition, 1925.
- [24] Richard Starý. *Matematická sazba*, 2005. <http://richardstary.wz.cz/>
- [25] Philip Taylor. Knižní úprava pro uživatele  $\TeX$ u. část druhá: Praxe. In *Zpravodaj  $\mathcal{CS}\mathcal{TUG}u$* , pages 38–60, 1995. Překlad Ladislav Šenkyřík.
- [26] Philip Taylor. Knižní úprava pro uživatele  $\TeX$ u. část první: Teorie. In *Zpravodaj  $\mathcal{CS}\mathcal{TUG}u$* , pages 20–37, 1995. Překlad Ladislav Šenkyřík.



- [27] Hàn Thế Thành. *Micro-typographic extensions to the T<sub>E</sub>X typesetting system*. PhD thesis, Masarykova univerzita v Brně, fakulta informatiky, 2000. Or: TUGboat 21,4, Dec 2000.
- [28] Hàn Thế Thành, Sebastian Rahtz, and Hans Hagen. *The pdfT<sub>E</sub>X user manual*. Pragma, 2001.  
<http://sarovar.org/docman/view.php/106/66/pdf-tex-s.pdf>
- [29] Zdeněk Wagner. Využití XML a L<sup>A</sup>T<sub>E</sub>Xu při sazbě odborných knih. *Zpravodaj Československého sdružení uživatelů T<sub>E</sub>Xu*, 13(1):188–211, 2002.
- [30] Norman Walsh and Leonard Mueller. *DocBook: The Definitive Guide*, 2006.  
<http://www.docbook.org/tdg/en/html/docbook.html>
- [31] Karel Wick. *Pravidla matematické sazby*. Academia, Praha, 1. edition, 1966.
- [32] Hermann Zapf. About micro-typography and the hz-program. *Electronic Publishing*, 6(3):283–288, 1993. <http://cajun.cs.nott.ac.uk/compsci/epo/papers/volume6/issue3/zapf.pdf>
- [33] Vít Zýka. Používáme pdfT<sub>E</sub>X IV: mikrotypografické rozšíření. *Zpravodaj Československého sdružení uživatelů T<sub>E</sub>Xu*, 14(2):47–53, November 2004.
- [34] Vít Zýka. Používáme pdfT<sub>E</sub>X IVa: hz-algoritmus jednodušeji. *Zpravodaj Československého sdružení uživatelů T<sub>E</sub>Xu*, 15(1):90–92, 2005.

## Summary: Documents preparation for typesetting

In this article we express the general principles of a good document and we pose the requirements for their editing, processing and visualisation. Based on these requirements we show that an appropriate format is a structure marked document. We explain what structure marking is and describe its features. Finally we mention the tools for manipulating with structure marked documents and we sketch the ways they are formatted by T<sub>E</sub>X.

V tomto tutoriálu si krok za krokem ukážeme, jak vytvořit článek pomocí CONTEXTu [3, 1, 2]. Výsledný text, viz obrázek 1, bude zkrácenou verzí skutečného odborného článku a bude tak obsahovat většinu prvků takového typu dokumentu.

## Zdrojový text tutoriálu

Strukturně značkový text torza článku:

```
\startproduct tutorial-paper-vzor

\environment paper-style-vzor

% ----- Title
\startTitle
\Title%
  Graph-based Range Image Registration\
  Combining Geometric and Photometric Features}
\Author%
  Ikuko Shimizu\Inst{1},
  Akihiro Sugimoto\Inst{2},
  Midar Asar\Inst{3}}
\Institute%
  \Inst{1}~Tokyo University of Agriculture and Technology, Japan\
  \Inst{2}~National Institute of Informatics, Japan\
  \Inst{3}~Czech Technical University, Czech Republic\
  \Email{ikuko@tt.tuat.ac.jp},
  \Email{sugimoto@nee.ac.jp},
  \Email{asar@ftr.cvut.cz}}
\stopTitle

% ----- Abstract
\startAbstract
We propose a coarse registration method of range images using both
geometric and photometric features. The framework of existing methods
using multiple features first defines a single similarity distance
summing up each feature based evaluations, and then minimizes the
distance between range images for registration. In contrast, we
formulate registration as a graph-based optimization problem, where we
independently evaluate geometric feature and photometric feature and
consider only the order of point-to-point matching quality. We then
find as large consistent matching as possible in the sense of the
matching-quality order. This is solved as one global combinatorial
optimization problem. Our method thus does not require any good
initial estimation and, at the same time, guarantees that the global
solution is achieved.
\stopAbstract
```

## Graph-based Range Image Registration Combining Geometric and Photometric Features

Riku Shimizu<sup>1</sup>, Akihiro Sugimoto<sup>2</sup>, Midar Asa<sup>3</sup>

<sup>1</sup> Tokyo University of Agriculture and Technology, Japan

<sup>2</sup> National Institute of Informatics, Japan

<sup>3</sup> Czech Technical University, Czech Republic  
rikushim@at.ac.jp, sugimoto@nii.ac.jp, asa@rrtr.cvut.cz

**Abstract.** We propose a coarse registration method of range images using both geometric and photometric features. The framework of existing methods using multiple features first defines a single similarity distance summing up each feature-based evaluations, and then minimizes the distance between range images for registration. In contrast, we formulate registration as a graph-based optimization problem, where we independently evaluate geometric feature and photometric feature and consider only the order of point-to-point matching quality. We then find as large consistent matching as possible in the sense of the matching-quality order. This is solved as one global combinatorial optimization problem. Our method thus does not require any good initial estimation and, at the same time, guarantees that the global solution is achieved.

### 1 Introduction

Automatic 3D model acquisition of the real-world object is important for many applications such as CAD/CAM or CG. A range sensor, which is a sensing device directly measuring 3D information of an object surface, is a useful tool in modeling 3D objects. An image of an object captured by a range sensor is called a range image and it provides a partial shape of the object in terms of the 3D coordinates of surface points in which the coordinate system is defined by the position and orientation of the range sensor. To obtain the full shape of an object, therefore, we have to align range images captured from different viewpoints. This alignment, i.e., finding the rigid transformation between coordinate systems that align given range images, is called range image registration.

Widely used methods for range image registration are the iterative closest point (ICP) method proposed by [1] and its extension [2]. These methods iterate two steps: Each point in one range image is transformed by a given transformation to find the closest point in the other range image. These point correspondences are then used to estimate the transformation minimizing matching errors. In order to robustly realize range image registration, some features reducing matching ambiguity are proposed in addition to simply computed geometric features. They are, for example, color attributes, chromaticity, normal vectors, curvatures themselves and their features, and attributes representing overlapping areas of planes. Combining different kinds of features enhances robustness for registration; nevertheless, defining one common nonrigid metric for similarity using different kinds of features is still even difficult.

<sup>1</sup> The term “robust” in this paper means that the possibility of successful registration is enhanced; registration is more successful.

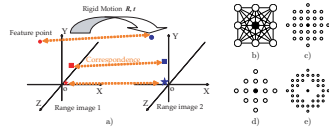
(a) Strana 1

On the other hand, a method using a graph-based optimization algorithm for range image registration is proposed. The method formalizes the matching problem as a discrete optimization problem in an oriented graph so that optimal matching becomes equivalent with the uniquely existing maximum strict sub-kernel (SSK) of the graph. As a result, this method does not require any good initial estimation and, at the same time, guarantees that the global solution is achieved. In addition, it also has an advantage that a part of data is reported rather than forcefully interpreted if evidence of correspondence is insufficient in the data or if it is ambiguous. The method, however, deals with geometric features only and fails in finding matching for data of an object having insufficient shape features.

In this paper, we extend the graph-based method so that it does work even for the case of data with insufficient shape features. We incorporate the combination of geometric and photometric features into the framework to enhance the robustness of registration. Existing methods combining such features define a single metric by adding or multiplying similarity criteria computed from each feature to find point matches. In contrast, our proposed method first evaluates each point match independently using each feature, and then determines the order of matching quality among all possible matches. To be more concrete, for two point-matches, if similarity of one match is greater than the other over all features, we regard that the former is strictly superior to the latter. Otherwise, we leave the order between the two matches undetermined. This is because both geometric and photometric features should be consistently similar with each other for a correct match. Introducing this partial order on matching quality to the graph-based method for range image registration allows us to find as large consistent matching with given data among all possible matches. The maximum SSK algorithm enables us to uniquely determine the largest consistent matching of points with guaranteeing the global solution. This indicates that our proposed method is useful for coarse registration.

### 2 Multiple features for reducing matching ambiguity

#### 2.1 3D point matching problem



**Figure 1.** Point-based registration of two range images (a) and augmented triangular mesh over  $3 \times 3$  vertex neighborhood. (b) 24 elementary triangles sharing the central vertex. (c) Local surface vertices neighborhood used to estimate a local normal vector from 32 triangles. (d) Neighborhood for computing the triple and photometric features (32 vertices). (e) Neighborhood for computing the triple feature (604 triangles).

(b) Strana 2

### 2.2 Employed features for registration

The features we will use are computed from the augmented triangular mesh which includes all possible triangles among triples of vertices in a small vertex neighborhood (Fig. 1b). We have chosen four local features, three of which are geometric and the other is photometric: (A) oriented surface normal, (B) structure matrix, (C) triple feature, and (D) chromaticity. We note that (A) and (B) are covariant features whereas (C) and (D) are invariant.

(A) Oriented surface normal. . . .

(B) Structure matrix. . . .

$$U^T p = RU, \quad D^T = D, \quad (1)$$

(C) Triple feature. . . .

### 2.3 Distribution based similarity evaluation

### 3 Graph-based registration method using multiple features

#### 3.1 Generating an unoriented graph $G$

#### 3.2 Generating an oriented graph $D$

#### 3.3 Strict sub-kernel of $D$

#### 4 Range image registration using SSK

#### 4.1 Maximum SSK and matching

Then, we use two sets of interest points, each of which is independently detected from one of two given range images, and generate a table for all possible matches. In generating the table, we eliminate matches that do not satisfy a given search range of rigid transformations. To be more concrete, for a given corresponding pair of points, we compute their structure matrices and then decompose them using SVD to find the rotation relating the pair (cf. Eq. (1)). Next, we eliminate the pair from the table if the rotation is not admissible.

### 5 Experiments

**Table 1.** Evaluation of registration results (“-” means failure in estimation).

points	1	2	3	4	5	6	7	7	9
IPs of 4th image	11610	10888	9913	9274	9442	9778	10063	11589	12118
IPs of 5th image	173	241	210	173	197	246	286	378	452
IPs of (4 + 5)-th image	207	229	172	193	209	243	190	171	172

(c) Strana 3

Table 1 shows that over all the cases, the registration accuracy of our method is not only significantly higher but also numerically more stable, compared with the method using geometric features alone.

### 6 Conclusion

We extended a graph-based range image registration method so that it can handle both geometric and photometric features simultaneously. Namely, we formulated registration as a graph-based optimization problem where we independently evaluate geometric feature and photometric feature and then consider only the order of point-to-point matching quality. We then find as large consistent matching as possible in the sense of the matching-quality order. This is solved as one global combinatorial optimization problem of polynomial complexity. The advantage of our method is that each match is independently evaluated by each employed feature and the order of matching-quality is only concerned. Differently from existing methods, our proposed method need not define any single metric of similarity for evaluating matching. Our experimental results demonstrate the effectiveness of our method for coarse registration.

The proposed method will reduce the possibility of finding an incorrect matching but cannot be expected to increase the number of matches significantly. This follows from the fact that both the two similarity criteria have to be consistent. In principle, it is also possible to combine the two criteria in such a way that when one of them strictly favors the match of  $q$  to  $p$  and the other is at least indifferent between  $p$  and  $q$ , the edge joining  $p$  and  $q$  becomes undirectional. Such definition requires using a different matching algorithm from the one used in this paper. This research direction is our ongoing work.

**Acknowledgments.** A part of this work was done under the framework of MOC between the Czech Technical University and National Institute of Informatics. This work is in part supported by the Czech Academy of Sciences under project IETI01210406 and by the EC project MITEN-CT-2004-065430.

### References

- [1] P. J. Besl and N. D. McKay. A method for registration of 3-D shapes. *IEEE Trans. on PAMI*, 14(2):239–256, 1992.
- [2] Y. Chen and G. Medioni. Object modeling by registration of multiple range images. *IVC*, 10(3):145–155, 1992.

(d) Strana 4

% ----- Introduction

\section{Introduction}

Automatic 3D model acquisition of the real-world object is important for many applications such as CAD/CAM or CG. A range sensor, which is a sensing device directly measuring 3D information of an object surface, is a useful tool in modeling 3D objects. An image of an object captured by a range sensor is called a range image and it provides a partial shape of the object in terms of the 3D coordinates of surface points in which the coordinate system is defined by the position and orientation of the range sensor. To obtain the full shape of an object, therefore, we have to align range images captured from different viewpoints. This alignment, i.e., finding the rigid transformation between coordinate systems that aligns given range images, is called range image registration.

Widely used methods for range image registration are the iterative closest point (ICP) method proposed by \cite{Besl92} and its extensions \cite{Chen92}. These methods iterate two steps: Each point in one range image is transformed by a given transformation to find the closest point in the other range image. These point correspondences are then used to estimate the transformation minimizing matching errors. In order to robustly\footnote{The terminology \bquote{robust} in this paper means that the possibility of successful registration is enhanced; registration is more successful.} realize range image registration, some features reducing matching ambiguity are proposed in addition to simply computed geometric features. They are, for example, color attributes, chromaticity, normal vectors, curvatures themselves and their features, and attributes representing overlapping areas of planes. Combining different kinds of features enhances robustness for registration; nevertheless, defining one common meaningful metric for similarity using different kinds of features is still even difficult.

On the other hand, a method using a graph-based optimization algorithm for range image registration is proposed. The method formalizes the matching problem as a discrete optimization problem in an oriented graph so that optimal matching becomes equivalent with the uniquely existing maximum strict sub-kernel (SSK) of the graph. As a result, this method does not require any good initial estimation and, at the same time, guarantees that the global solution is achieved. In addition, it also has an advantage that a part of data is rejected rather than forcefully interpreted if evidence of correspondence is insufficient in the data or if it is ambiguous. The method, however, deals with geometric features only and fails in finding matching for data of an object having insufficient shape features.

In this paper, we extend the graph-based method so that it does work even for the case of data with insufficient shape features. We incorporate the combination of geometric and photometric features into the framework to enhance the robustness of registration. Existing methods combining such features define a single metric by adding or multiplying similarity criteria computed from each feature to find point matches. In contrast, our proposed method first evaluates each point match independently using each feature, and then determines the order of matching quality among all possible matches. To be more concrete, for two point-matches, if similarity of one match is greater than the other over all features, we regard that the former is strictly

superior to the latter. Otherwise, we leave the order between the two matches undetermined. This is because both geometric and photometric features should be consistently similar with each other for a correct match. Introducing this partial order on matching quality to the graph-based method for range image registration allows us to find as large consistent matching with given data among all possible matches. The maximum SSK algorithm enables us to uniquely determine the largest consistent matching of points with guaranteeing the global solution. This indicates that our proposed method is useful for coarse registration.

```
% ----- ...
\section{Multiple features for reducing matching ambiguity}

\subsection{3D point matching problem}
\placefigure
[here,top]
[fig:p-to-p]
{Point-based registration of two range images
(a)~and augmented triangular mesh over  $3 \times 3$  vertex
neighborhood.
(b)~24 elementary triangles sharing the central vertex.
(c)~Local surface vertices neighborhood used to estimate a
local normal vector from 332 triangles.
(d)~Neighborhood for computing the triple and photometric features
(52 vertices).
(e)~Neighborhood for computing the triple feature (604 triangles).}
{\tmpD=.12\textwidth
\hbox{%
\startcombination[1*1]
{\externalfigure[correspondence][width=.55\textwidth]}{a}}
\stopcombination
\hskip1cm
\startcombination[2*2]
{\externalfigure[3x3trg][width=\tmpD]}{b}}
{\externalfigure[7x7ngh][width=\tmpD]}{c}}
{\externalfigure[5x5ngh][width=\tmpD]}{d}}
{\externalfigure[7x7onion2][width=\tmpD]}{e}}
\stopcombination
}}

```

```
\subsection{Employed features for registration}
The features we will use are computed from the augmented triangular
mesh which includes all possible triangles among
triples of vertices in a small vertex neighborhood
(\in{Fig.}{b}[fig:p-to-p]). We have chosen four local features, three
of which are geometric and the other is photometric: (A) oriented
surface normal, (B) structure matrix, (C) triple feature, and (D)
chromaticity. We note that (A) and (B) are covariant features whereas
(C) and (D) are invariant.
```

```
%\noindent{\bf (A) Oriented surface normal.}
\subject{(A) Oriented surface normal}\dots
```

```
\subject{(B) Structure matrix}\dots
```

```
\placeformula[eq:covariant-f]
$$
\mat{U}'\mat{P} = \mat{R}\mat{U}, \quad \quad
\mat{D}' = \mat{D} \quad \backslash:,
$$
```

```

\subject{(C) Triple feature}\dots
\subsection{Distribution based similarity evaluation}

% ----- ...
\section{Graph-based registration method using multiple features}
\subsection{Generating an unoriented graph  $\mathcal{G}$ }
\subsection{Generating an oriented graph  $\mathcal{D}$ }
\subsection{Strict sub-kernel of  $\mathcal{D}$ }
% ----- ...
\section{Range image registration using SSK}
\subsection{Maximum SSK and matching}
Then, we use two sets of interest points, each of which is
independently detected from one of two given range images, and
generate a table for all possible matches. In generating the table, we
eliminate matches that do not satisfy a given search range of rigid
transformations. To be more concrete, for a given corresponding pair
of points, we compute their structure matrices and then decompose them
using SVD to find the rotation relating the pair
(cf. \inM[eq:covariant-f]). Next, we eliminate the pair from
the table if the rotation is not admissible.

% ----- ...
\section{Experiments}

\start
\setupcaptions[align=middle,location=top]
\setuptables[bodyfont=8pt]
\placetable
[bottom,top]
[tab:s-match-tab]
{Evaluation of registration results (“-$” means failure in estimation).}
{\starttable[|l|l|r|r|r|r|r|r|r|r|]
%HL
\NC \NC  $i$  \NC 1 \NC 2 \NC 3 \NC 4 \NC 5 \NC 6 \NC 7 \NC 7 \NC 9 \NC \SR
\HL
\NC \NC points \NC
11616 \NC 10888 \NC 9913 \NC 9374 \NC 9442 \NC 9778 \NC
10503 \NC 11589 \NC 12118 \NC \FR
\NC \NC IPs of  $i$ -th image \NC
173 \NC 241 \NC 210 \NC 173 \NC 197 \NC 210 \NC 260 \NC 178 \NC 172 \NC \MR
\NC \NC IPs of  $(i+1)$ -th image \NC
237 \NC 229 \NC 172 \NC 193 \NC 269 \NC 243 \NC 190 \NC 171 \NC 172 \NC \LR
%HL
\stoptable}
\stop

\in{Table}{}[tab:s-match-tab] shows that over all the cases,
the registration accuracy of our method is not only significantly
higher but also numerically more stable, compared with the method
using geometric features alone.

% ----- Conclusion
\section{Conclusion}
We extended a graph-based range image registration method so that it
can handle both geometric and photometric features simultaneously.
Namely, we formulated registration as a graph-based optimization
problem where we independently evaluate geometric feature and
photometric feature and then consider only the order of point-to-point
matching quality. We then find as large consistent matching as

```

possible in the sense of the matching-quality order. This is solved as one global combinatorial optimization problem of polynomial complexity. The advantage of our method is that each match is independently evaluated by each employed feature and the order of matching-quality is only concerned. Differently from existing methods, our proposed method need not define any single metric of similarity for evaluating matching. Our experimental results demonstrate the effectiveness of our method for coarse registration.

The proposed method will reduce the possibility of finding an incorrect matching but cannot be expected to increase the number of matches significantly. This follows from the fact that both the two similarity criteria have to be consistent. In principle, it is also possible to combine the two criteria in such a way that when one of them strictly favors the match of  $q$  to  $p$  and the other is at least indifferent between  $p$  and  $q$ , the edge joining  $p$  and  $q$  becomes unidirectional. Such definition requires using a different matching algorithm from the one used in this paper. This research direction is our ongoing work.

**\subject{Acknowledgments}**

A part of this work was done under the framework of MOU between the Czech Technical University and National Institute of Informatics. This work is in part supported by the Czech Academy of Sciences under project 1ET101210406 and by the EC project MRTN-CT-2004-005439.

% ----- Bibliography

**\iffalse** % commented; we use the bib file instead

**\startpublication**

[k=Bes192,t=article,a=Bes1,y=1992,s=,n=,u=]

**\artauthor**[P.-J.]{Bes1}

**\artauthor**[N.-D.]{McKay}

**\arttitle**{A Method for Registration of 3-D Shapes}

**\journal**{IEEE Trans. on PAMI}

**\pubyear**{1992}

**\volume**{14}

**\issue**{2}

**\pages**{239--256}

**\stoppublication**

**\startpublication**

[k=Chen92]

**\artauthor**{Y.}{Chen}

**\artauthor**{G.}{Medioni}

**\arttitle**{Object Modeling by Registration of Multiple Range Images}

**\journal**{IVC}

**\pubyear**{1992}

**\volume**{10}

**\issue**{3}

**\pages**{145--155}

**\stoppublication**

**\fi**

**\completepublications**

**\stopproduct**

Definice strukturních značek, vzhledu a formátování:

```
\startenvironment paper-style-vzor

% --- Layout
\setuplayout[]
\setuppagenumbering[alternative=doublesided,location=]

% --- Font
\setupbodyfont[10pt]
\setupindenting[medium,yes]

% --- Header
\setupheadertexts[][pagenumber][pagenumber][] % odd(left,right) even(left,right)

% --- Title
\def\startTitle{%
  \startalignment[middle]
}
\def\stopTitle{%
  \stopalignment
}

\def\Title#1{{\bfc\setupinterlinespace#1\blank}}
\def\Author#1{#1\blank[2*big]}
\def\Inst#1{\high{#1}}

\def\Institute#1{{\let\\\crlf#1\blank[4*big]}}
\def>Email#1{{\tt#1}}

% --- Abstract
\def\startAbstract
  {\startnarrower
   {\bf Abstract. }%
  }
\def\stopAbstract
  {\stopnarrower
  }

% --- Heads
\setuphead
  [section]
  [style=\bfb]

\setuphead
  [title]
  [page=no,style=\bfb]

\setuphead
  [subject]
  [page=no,style=\bf,before=\blank,after=,alternative=text,textcommand=\putDot]
\def\putDot#1{#1.}

% --- Bibliography
\usemodule[bib]
\usemodule[bibltx]
\setupbibtex[database=tutorial-paper-vzor]
\setuppublications[alternative=num,autohang=yes]

% --- Math
\def\mat#1{{\bf#1}}
```



```

\def\inM[#1]{\setupreferencing[separator=]\in{Eq.~(}{)}[#1]}
% --- Figs
\setupexternalfigures[directory={./figs}]
\setupcaptions[width=\textwidth,stopper=.]
% --- Links
% --- Misc
\newdimen\tmpD
\stopenvironment

```

---

Bib<sub>T</sub><sub>E</sub><sub>X</sub> databáze citované literatury:

```

@Article{Besl92,
  author = {Besl, P. J. and McKay, N. D.},
  title = {A Method for Registration of {3-D} Shapes},
  journal = {IEEE Trans. on PAMI},
  year = {1992},
  volume = {14},
  number = {2},
  pages = {239--256},
}

@Article{Chen92,
  author = {Y. Chen and G. Medioni},
  title = {Object Modeling by Registration of Multiple Range Images},
  journal = {IVC},
  year = {1992},
  volume = {10},
  number = {3},
  pages = {145--155},
}

@Book{Pattern,
  author = {Duda, R. O. and Hart, P. E. and Stork, D. G.},
  title = {Pattern Classification},
  publisher = {John Willey and Sons, Inc.},
  year = {2001},
  edition = {2},
}

```

Zdrojové kódy tutoriálu jsou ke stažení na adrese <http://zyka.net>.

## Reference

- [1] Hans Hagen. *ConT<sub>E</sub>Xt the manual*. <http://www.pragma-ade.com/general/manuals/cont-eni.pdf>, November 2001.
- [2] Taco Hoekwater. *ConT<sub>E</sub>Xt module: Bibliographies*, 2006. <http://dl.contextgarden.net/modules/t-bib/doc/context/bib/bibmod-doc.pdf>.

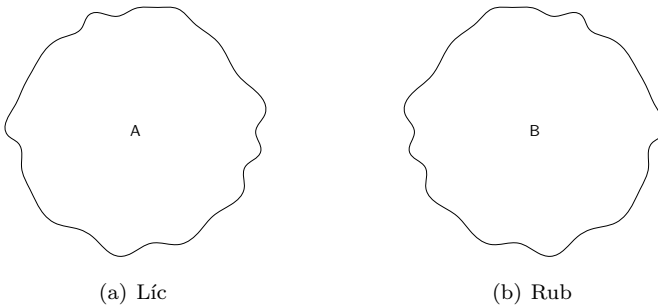
- [3] Ton Otten and Hans Hagen. Exkurze do ConT<sub>E</sub>Xtu. *Zpravodaj Československého sdružení uživatelů T<sub>E</sub>Xu*, 16(2–4):59–224, 2006. Český překlad manuálu ConT<sub>E</sub>Xt on Excursion, <https://foundry.supelec.fr/docman/view.php/22/14/ma-cb-cz-screen.pdf>.

## Summary: Article by ConT<sub>E</sub>Xt: Tutorial

In this tutorial we show how to create a technical article using ConT<sub>E</sub>Xt [3, 1, 2]. The resulting text will be a shortened version of the real article, see Figure 1, and so it will contain most of the elements of this kind of document.

*Vít Zýka* [vit.zyka@seznam.cz](mailto:vít.zyka@seznam.cz)

V tomto tutoriálu si ukážeme, jak vytvořit vektorový obrázek v PDF pomocí ConTEXtu. Výsledný obrázek, viz obrázek 1, bude dosti primitivní, ale ilustrativní. Kreslit budeme jazykem METAPOSTu, ale jeho popis není naším cílem. Tím bude krok za krokem z čistého stolu ukázat infrastrukturu, kterou nám ConTEXt k takové práci nabízí [1].



Obrázek 1: Výsledný obrázek tohoto tutoriálu autor použil pro tvorbu „pečetí“ na studentské imatrikulační glejty. Místo centrálního písmene bylo na líci použito logo gymnázia, na rubu označení třídy.

## Zdrojový text tutoriálu

```
\setupbodyfont[ss,10pt]
\setupexternalfigures[location=global]
\setuplayout
  [backspace=8mm, topspace=8mm, bottomspace=8mm,
  header=0pt, footer=0pt,
  width=fit, height=fit]
\noheaderandfooterlines
\def\Pecet{\dosingleempty\doPecet}
\def\doPecet[#1]#2{%
  \setupMPvariables[SisoidGraphic][seed=1, reflect=false, #1]
  \startuseMPgraphic{SisoidGraphic}
```

```

N := 10; da := 3; dr := 2mm;
%N := 20; da := 8; dr := 10mm; % shape variant
randomseed := \MPvar{seed};
path P;

def RandAngle = (da*(uniformdeviate(2)-1)) enddef;
def RandRadius = (dr*(uniformdeviate(2)-1)) enddef;

def Point(expr a) =
  begingroup
  aa := a + RandAngle;
  x := R + ((R+RandRadius)*cosd(aa));
  y := R + ((R+RandRadius)*sind(aa));
  (x,y)
endgroup
enddef;

R := .5 * .9 * \overlaywidth;
P := (
  for a:=0 step N until (360-N):
    Point(a)..
  endfor
  cycle)
if \MPvar{reflect}: reflectedabout((R,R), (R,R+R)) fi;
draw P;
\stopuseMPgraphic
\defineoverlay[SisoidOverlay][\useMPgraphic{SisoidGraphic}]%
\SisoidFrame{#2}%
}

\defineframed[SisoidFrame]
[width=60mm, height=60mm, background=SisoidOverlay, frame=off]

\def\PecetLic[#1]{\Pecet[reflect=false,#1]{\bfd A}}
\def\PecetRub[#1]{\Pecet[reflect=true, #1]{\bfd B}}

\starttext
\startTEXpage
\Pecet{A}%
\stopTEXpage

\startTEXpage
\Pecet[reflect=true]{B}%
\stopTEXpage

\leavevmode\dorecurse{12}{\PecetLic[seed=#1]}%
\page
\leavevmode\dorecurse{12}{\PecetRub[seed=#1]}%
\stoptext

```

Zdrojové kódy tutoriálu jsou ke stažení na adrese <http://zyka.net>.

## Reference

- [1] Hans Hagen. *META FUN*. 2002.  
<http://www.pragma-ade.com/general/manuals/metafun-s.pdf>

## Summary: Logo by ConT<sub>E</sub>Xt: Tutorial

In this tutorial we are showing how to create a PDF vector figure by ConT<sub>E</sub>Xt. The figure, see Figure 1, is rather primitive but illustrative. Although the drawing uses METAPOST its language description is not our goal. We are focusing on a step-by-step demonstration of ConT<sub>E</sub>Xt infrastructure for this kind of work [1].

*Vít Zýka* `vit.zyka@seznam.cz`

Sázecího systému  $\text{\TeX}$  lze s velkým prospěchem využívat také v klasické filologii. I zde totiž existují speciální požadavky na sazbu, o jejichž realizaci si uživatel nejrozšířenějších textových procesorů typu WYSIWYG mohou nechat jenom zdát. Jedním z takových požadavků je tzv. zrcadlová sazba. Klasičtí filologové často chtějí, aby jejich překlady řeckých a latinských textů byly vydány v jedné knize paralelně s textem originálu: Na sudých (levých) stranách bývá originál, na lichých (pravých) překlad; přitom na každé dvojstraně musí být stejný úsek originálu a překladu. Někdy se na každou lichou stranu přímo pod text překladu připojují překladatelské poznámky, které se vztahují k místům překladu na téže straně. Na sudých stranách může být pod textem originálu kritický aparát, obsahující textové varianty.

V tomto článku bych chtěl ukázat jedno z možných řešení zrcadlové sazby.<sup>1</sup> Nepůjde nám o to, připravit konkrétní vzhled sazby konkrétního překladu a jeho originálu, nýbrž vytvořit skupinu maker, jejichž vhodným užitím bude moci šikovný  $\text{\TeX}$ ista pohodlně pořizovat zrcadlovou sazbu i značně komplikovaných dokumentů; vzhled sazby bude přitom plně v jeho rukou. Naše makra budou určena pro plain a příbuzné formáty. Důraz budeme klást na to, aby celá sazba probíhala automaticky a aby bylo možno užívat insertů obvyklým způsobem.

Základní myšlenka je tato: Každý z obou textů<sup>2</sup> bude uložen v jiném souboru; levý text načteme do jednoho  $\text{\vbox}$ u, pravý text do druhého  $\text{\vbox}$ u a potom budeme z těchto  $\text{\vbox}$ ů pomocí operace  $\text{\vsplit}$  postupně odkrajovat jednotlivé strany.

Kdybychom se však pokusili načíst oba texty naráz, zřejmě by došlo k překročení paměťových možností  $\text{\TeX}$ u. Budeme je tedy načítat po částech tak, aby v obou  $\text{\vbox}$ ech (nazývávejme je  $\text{\levybox}$  a  $\text{\pravybox}$ ) bylo stále dost materiálu na odkrojení jedné strany. Doplnění  $\text{\vbox}$ ů bude probíhat v podstatě takto: Ze zdrojového souboru budeme číst jednotlivé řádky pomocí příkazu  $\text{\read}$ ; každý přečtený řádek zkopírujeme do pomocného souboru. Jakmile narazíme na prázdný řádek (tedy zhruba řečeno na konec odstavce), přerušíme čtení a kopírování řádků, pomocí příkazu  $\text{\input}$  načteme pomocný soubor do příslušného  $\text{\vbox}$ u a poté přeměříme výšku tohoto  $\text{\vbox}$ u. Uvedený cyklus budeme opakovat tak dlouho, dokud výška  $\text{\vbox}$ u nebude mít hodnotu alespoň jeden a čtvrt  $\text{\vsize}$  nebo dokud nedosáhneme konce zdrojového souboru.

<sup>1</sup>Poznamenejme, že jiné řešení stejného úkolu bylo popsáno ve Zpravodaji č. 4 z roku 1997.

<sup>2</sup>To jest originál a překlad. Dále budeme oba texty označovat obecněji podle jejich polohy na dvojstraně jako „levý text“ a „pravý text“.

Občas se stává, že nějaká skupina přesahuje z jednoho odstavce do druhého; dochází k tomu např. při sazbě uvozovek pomocí makra `\uv`. V takovém případě je nežádoucí, aby načítání textu bylo přerušeno mezi těmito dvěma odstavci. Uživatel by samozřejmě mohl ošetřit takové případy tím, že by uvnitř skupin používal místo prázdného řádku řídicí sekvenci `\par`. Aby se uživatel nemusel tímto problémem vůbec zabývat, vytvoříme mechanismus, který se sám postará o to, aby načítání textu nebylo přerušeno uvnitř skupiny. Přitom budeme mít na zřeteli, že skupiny, uvnitř kterých se mohou vyskytnout prázdné řádky, nemusejí být ohraničeny jen explicitními kudrnatými závorkami, nýbrž že mohou být vymezeny také pomocí implicitních závorek alias `\bgroup` a `\egroup`, dále pomocí alternativních příkazů `\begingroup` a `\endgroup`, případně pomocí nějakých uživatelem definovaných dvojic maker.

Teď, když už víme, co by měla umět makra, která budou zajišťovat doplňování boxů `\levybox` a `\pravybox`, pustíme se do jejich definování.

```

1 \newread\levysoubor % zdrojový soubor s levým textem
2 \newbox\levybox % box, kam budeme načítat levý text
3 \newdimen\prevdepthL % \prevdepth na konci boxu \levybox
4 \let\nastavlevy=\relax % specifická nastavení pro levý text
5 \def\doplNlevybox{\let\next=\relax
6   \ifdim\ht\levybox<1.25\vsiz
7     \ifeof\levysoubor \else \nactilevytext
8       \let\next=\doplNlevybox \fi\fi \next}
9 \def\nactilevytext{\let\soubor=\levysoubor \kopiruj
10  \setbox\levybox=\vbox{\break \unvbox\levybox
11    \let\versnum=\versnumL \prevdepth=\prevdepthL
12    \nastavlevy \input\jobname.tmp
13    \global\prevdepthL=\prevdepth}%
14  \setbox0=\vsplit\levybox to Opt }
```

V rámci cyklu `\doplNlevybox` pracuje makro `\nactilevytext`. Z něj je nejdříve zavoláno makro `\kopiruj`, které zkopíruje ze zdrojového souboru do pomocného souboru úsek textu ohraničený prázdným řádkem; ke kódu tohoto makra se dostaneme za chvíli. Pomocný soubor je potom načten do boxu `\levybox`. Nastavení `\prevdepth` na hodnotu, která byla při minulém načítání zaznamenána do registru `\prevdepthL`, zajistí správnou meziřádkovou mezeru mezi odstavci. Řídicí sekvence `\nastavlevy`, která se aktivuje vždy před načtením pomocného souboru, je určena pro uživatele; uživatel ji může nadefinovat jako makro, které provede specifická nastavení pro levý text (nastavení správné tabulky dělení slov apod.). Řídicí sekvence `\versnum` a `\versnumL` souvisejí se synchronizací textů; o jejich přesném významu si povíme později. Nakonec jsou pomocí operace `\vsplit` vymazány ze začátku boxu `\levybox` odstranitelné elementy a místo nich je vložena mezeru podle `\splittopskip`.

Obdobně definujeme makra \doplňpravybox a \nactipravytext:

```
15 \newread\pravysoubor % zdrojový soubor s pravým textem
16 \newbox\pravybox % box, kam budeme načítat pravý text
17 \newdimen\prevdepthP % \prevdepth na konci boxu \pravybox
18 \let\nastavpravy=\relax % specifická nastavení pro pravý text
19 \def\doplňpravybox{\let\next=\relax
20   \ifdim\ht\pravybox<1.25\vszise
21     \ifeof\pravysoubor \else \nactipravytext
22       \let\next=\doplňpravybox \fi\fi \next}
23 \def\nactipravytext{\let\soubor=\pravysoubor \kopiruj
24   \setbox\pravybox=\vbox{\break \unvbox\pravybox
25     \let\versnum=\versnumP \prevdepth=\prevdepthP
26     \nastavpravy \input\jobname.tmp
27     \global\prevdepthP=\prevdepth}%
28   \setbox0=\vsplit\pravybox to Opt }
```

Zde je kód makra \kopiruj spolu s kódem několika dalších maker, která s ním úzce souvisejí:

```
29 \newwrite\tempfile % pomocný soubor
30 \newcount\groupnum % počet otevřených skupin
31 \def\kopiruj{\begingroup \endlinechar=-1 \deaktivuj
32   \aktivniznaky \nullfont \tracinglostchars=0
33   \immediate\openout\tempfile=\jobname.tmp
34   \let\next=\ctiradek \ctiradek
35   \immediate\closeout\tempfile \endgroup}
36 \def\ctiradek{\read\soubor to\temp
37   \immediate\write\tempfile{\temp}%
38   \setbox0=\hbox{\temp\relax}%
39   \ifx\temp\empty \ifnum\groupnum=0 \let\next=\relax \fi\fi
40   \next}
41 \def\deaktivuj{\catcode'\ =12 \catcode'\$=12 \catcode'\&=12
42   \catcode'\#=12 \catcode'\^=12 \catcode'\_ =12 \catcode'\~ =12 }
43 \begingroup
44 \catcode'\|=0 \catcode'\[=1 \catcode'\]=2 \catcode'\X=14
45 \catcode'\{=13 \catcode'\}=13 \catcode'\%=13 \catcode'\|=13
46 |gdef|aktivniznaky[X
47   |catcode'\{=13 |catcode'\}=13 |catcode'\%=13 |catcode'\|=13
48   |def|[|string{|ifhmode|global|advance|groupnum by 1 |fi|X
49   |def|[|string|ifhmode|global|advance|groupnum by -1 |fi|X
50   |def%|[|string%|ifhmode|expandafter|skryjkomentar|fi|X
51   |def|[|string\|ifhmode|def|temp[]|expandafter|ctitoken|fi]]
52 |endgroup
```



```

53 \def\skryjkomentar#1\relax{}
54 \def\ctitoken#1{%
55   \ifcat\noexpand#1a\edef\temp{\temp#1}\let\next=\ctitoken
56   \else \ifx\temp\empty \def\temp{#1}\let\next=\relax
57   \else \def\next{#1}\fi \testgroup \fi \next}
58 \def\idbgroup{bgroup} % identifikátor říd. sekvence \bgroup
59 \def\idegroup{egroup} % identifikátor říd. sekvence \egroup
60 \def\idbegingroup{begingroup} % ident. říd. sekv. \begingroup
61 \def\idendgroup{endgroup} % ident. říd. sekv. \endgroup
62 \def\testgroup{%
63   \ifx\temp\idbgroup \global\advance\groupnum by 1 \fi
64   \ifx\temp\idegroup \global\advance\groupnum by -1 \fi
65   \ifx\temp\idbegingroup \global\advance\groupnum by 1 \fi
66   \ifx\temp\idendgroup \global\advance\groupnum by -1 \fi}

```

Makro `\kopiruj` nejdříve pomocí záporné hodnoty registru `\endlinechar` sdělí input procesoru, aby na konec řádků nepřipojoval žádný znak. Dále znakům se speciálním významem přiřadí kategorii 12; výjimkou jsou kudrnaté závorky, zpětné lomítko a procento — ty se stanou aktivními znaky. Při zápisu do pomocného souboru (tj. mimo horizontální mód) expandují pouze na token kategorie 12. Ostatní jejich činnost bude probíhat v pracovním `\hboxu` (viz řádek 38): Levá kudrnatá závorka zvýší o jedničku hodnotu registru `\groupnum`, zatímco pravá kudrnatá závorka hodnotu registru `\groupnum` o jedničku sníží. Zpětné lomítko uloží identifikátor řídicí sekvence do těla definice makra `\temp` a poté prostřednictvím makra `\testgroup` rozhodne, zda se hodnota registru `\groupnum` zvýší, sníží, anebo zůstane beze změny. V makru `\testgroup` se testuje, zda identifikátor uložený do těla definice makra `\temp` je shodný s identifikátorem některé ze řídicích sekvencí `\bgroup`, `\egroup`, `\begingroup` a `\endgroup`. Podle potřeby je samozřejmě možno rozšířit makro `\testgroup` o další testy. Procento se postará o to, aby případné začátky a konce skupin, které se objeví v komentáři, neovlivnily hodnotu registru `\groupnum`.

Užití prázdného fontu `\nullfont`, který neobsahuje žádné znaky, mírně urychlí vytváření horizontálního seznamu uvnitř pracovního `\hboxu`; musíme ovšem zároveň nastavit `\tracinglostchars` na nulu, aby nebyla do souboru log vypisována hlášení o chybějících znacích, jinak by se naopak činnost  $\TeX$ u zpomalila.<sup>3</sup>

Přečtení jednoho řádku ze zdrojového souboru, zapsání tohoto řádku do pomocného souboru a vypuštění téhož řádku do pracovního `\hboxu` bude v rámci

<sup>3</sup>V dnešní době jsou počítače natolik rychlé, že mírné zrychlení, jakého lze dosáhnout užitím prázdného fontu, se jeví jako takřka bezvýznamné. Pozorný čtenář si navíc mohl všimnout toho, že naše makra nejsou optimalizována z hlediska rychlosti zpracování, nýbrž že jsme se snažili o co možná nejlépe čitelný kód. Zde jsme ale chtěli poukázat na existenci primitivu `\nullfont` a na jeho možné využití v praxi.

makra `\ctiradek` opakováno pořád dokola. Tento cyklus bude ukončen tehdy, když bude ze zdrojového souboru přečten prázdný řádek a současně hodnota registru `\groupnum` bude rovna nule. V makru `\ctiradek` není test, zda bylo dosaženo konce zdrojového souboru. Přesto bude cyklus na konci zdrojového souboru řádně ukončen i v případě, že uživatel nenechá na konci zdrojového souboru prázdný řádek.  $\TeX$  totiž automaticky připojuje prázdný řádek na konec každého souboru, který je otevřen pomocí příkazu `\openin`.

Nyní se dostáváme k problematice synchronizace obou textů. Nejdříve si nastíníme postup řešení této problematiky; k tomu nám pomůže následující úvaha. Otevřeme-li si bibli, již po zbežném zhlédnutí jedné strany poznáme, že text je rozčleněn do poměrně krátkých úseků, nazývaných „verše“.<sup>4</sup> Hned by nás mohlo napadnout, že při zrcadlovém vydání řeckého originálu a českého překladu Nového zákona by bylo vhodné, aby na každé dvojstraně byly stejné verše originálu i překladu. Tuto myšlenku rozvineme dále, neboť víme, že prakticky všechny řecké a latinské texty jsou členěny podobným způsobem a že většina překladatelů vyznačuje ve svých překladech členění korespondující s členěním originálu. Představme si tedy, že na začátku každého verše je pomocí příkazu `\mark` vložena značka, jejíž obsah udává pořadí tohoto verše v celém textu (průběžné číslování veršů je samozřejmě provedeno pro levý a pravý text zvlášť). Za takové situace můžeme pomocí příkazů `\vsplit to\vszie` a `\splitbotmark` zjistit, které verše se vejdou na levou stranu právě sázené dvojstrany a které verše na stranu pravou. Pokud by se ukázalo, že číslo posledního verše v materiálu určeném pro jednu stranu je vyšší než číslo posledního verše v materiálu určeném pro protilehlou stranu, můžeme odkrojený materiál, na jehož konci je verš s vyšším číslem, vrátit do příslušného `\vboxu` a z tohoto `\vboxu` odkrojit materiál o výšce menší než je `\vszie` tak, aby číslo posledního verše v tomto materiálu bylo shodné s číslem posledního verše v materiálu, který jsme odkrojili pro protilehlou stranu.

Implementace synchronizačních značek by mohla vypadat takto: Zařídíme, aby v souboru s levým i pravým textem byla na začátku každého verše zapsána řídicí sekvence `\vers`, a do souboru maker přidáme tyto řádky:

```
67 \newcount\versnumL \newcount\versnumP
68 \def\vers{\global\advance\versnum by 1
69   \leavevmode\mark{\the\versnum:}}
```

V definici makra `\vers` jsme využili toho, že se při načítání souborů mění význam řídicí sekvence `\versnum` podle toho, jestli se právě načítá levý, nebo pravý text (viz řádky 11 a 25).

Prolistujeme-li si více zrcadlových vydání řeckých a latinských textů a jejich překladů, zjistíme, že se často v záhlaví objevuje číslo (přesněji řečeno označení)

---

<sup>4</sup>Nenechme se zmást: Pojem „verš“ tu neznamená totéž, co se jako verš označuje v poezii. Dalo by se spíše říci, že verš zde znamená asi tolik co „věta“. Slůvko „asi“ je důležité, neboť oba pojmy — „verš“ a „věta“ — nelze plně ztotožnit.

prvního a posledního verše na aktuální straně. K tomuto účelu se v  $\TeX$  využívá značek typu `mark`.  $\TeX$  přitom disponuje jedinou třídou těchto značek, a tu jsme právě vypotřebovali na synchronizační značky; je však jasné, že číslo vložené do synchronizační značky se vůbec nemusí shodovat s konvenčním označením verše. Navíc bychom pro sazbu záhlaví potřebovali znát nejen označení veršů, ale také označení kapitol apod. Do jednotlivých značek typu `mark` budeme tedy muset vměstnat více informací.

Prostor v každé značce typu `mark` rozdělíme na dva oddíly: První oddíl bude obsahovat synchronizační číslo získané z registru `\versnumL`, resp. `\versnumP`; druhý oddíl bude mít plně k dispozici uživatel. Pokud bude uživatel chtít do tohoto oddílu vložit nějakou informaci, místo makra `\vers` užije makra `\Vers`.

```
70 \def\Vers#1{\global\advance\versnum by 1
71   \leavevmode\mark{\the\versnum:#1:}}
```

Následující makra nám zpřístupní informace ze značek typu `mark`:

```
72 \newcount\botversnumL
73 \def\readmarksL{%
74   \expandafter\doreadfirstL\splitfirstmark
75   \expandafter\doreadbotL\splitbotmark}
76 \def\doreadfirstL#1:#2:{\def\firstmarkL{#2}}
77 \def\doreadbotL#1:#2:{\botversnumL=#1 \def\botmarkL{#2}}
78 \newcount\botversnumP
79 \def\readmarksP{%
80   \expandafter\doreadfirstP\splitfirstmark
81   \expandafter\doreadbotP\splitbotmark}
82 \def\doreadfirstP#1:#2:{\def\firstmarkP{#2}}
83 \def\doreadbotP#1:#2:{\botversnumP=#1 \def\botmarkP{#2}}
```

Makro `\readmarksL` uloží do registru `\botversnumL` synchronizační číslo získané z poslední značky typu `mark` na levé straně; dále do těla definice maker `\firstmarkL` a `\botmarkL` uloží obsah uživatelského oddílu z první a poslední značky typu `mark` na levé straně. Obdobně funguje makro `\readmarksP`, které je ovšem určeno pro pravé strany. S makry `\firstmarkL`, `\botmarkL`, `\firstmarkP` a `\botmarkP` může uživatel pracovat ve výstupní rutině stejným způsobem, jako se běžně pracuje s primitivou `\firstmark` a `\botmark`.

V případě, že by se náhodou na nějaké straně neobjevila žádná značka typu `mark`, budeme pracovat s obsahem poslední značky na předcházející straně; k tomu slouží tyto definice:

```
84 \def\readlastpagemarkL{%
85   \expandafter\doreadfirstL\lastpagemarkL
86   \expandafter\doreadbotL\lastpagemarkL
87   \let\lastsplitmarkL=\lastpagemarkL}
```

```

88 \def\readlastpagemarkP{%
89   \expandafter\doreadfirstP\lastpagemarkP
90   \expandafter\doreadbotP\lastpagemarkP
91   \let\lastsplitmarkP=\lastpagemarkP}

```

Teď si ukážeme definice maker, která budou z boxů `\levybox` a `\pravybox` odkrajovat jednotlivé strany.

```

92 \newbox\levastrana
93 \newbox\zalohalevybox
94 \newdimen\vsizel
95 \def\odkrojlevou{\setbox\zalohalevybox=\copy\levybox
96   \ifdim\vsizel>0pt
97     \setbox\levastrana=\vsplit\levybox to\vsizel
98     \edef\lastsplitmarkL{\splitbotmark}%
99     \ifx\lastsplitmarkL\empty \readlastpagemarkL
100     \else \readmarksL \fi
101   \else
102     \setbox\levybox=\vbox{\line{} \nobreak \vskip-\topskip
103       \nointerlineskip\line{} \unvbox\levybox}%
104     \setbox\levastrana=\vsplit\levybox to-\topskip
105     \readlastpagemarkL
106   \fi}
107 \newbox\pravastrana
108 \newbox\zalohaprawybox
109 \newdimen\vsizelP
110 \def\odkrojpravou{\setbox\zalohaprawybox=\copy\pravybox
111   \ifdim\vsizelP>0pt
112     \setbox\pravastrana=\vsplit\pravybox to\vsizelP
113     \edef\lastsplitmarkP{\splitbotmark}%
114     \ifx\lastsplitmarkP\empty \readlastpagemarkP
115     \else \readmarksP \fi
116   \else
117     \setbox\pravybox=\vbox{\line{} \nobreak \vskip-\topskip
118       \nointerlineskip\line{} \unvbox\pravybox}%
119     \setbox\pravastrana=\vsplit\pravybox to-\topskip
120     \readlastpagemarkP
121   \fi}

```

Předpokládejme, že výchozí hodnota `\vsizel` i `\vsizelP` je rovna `\vsizel`. Makra `\odkrojlevou` a `\odkrojpravou` odkrojí z boxů `\levybox` a `\pravybox` potenciální levou a pravou stranu, tyto dvě potenciální strany uloží do boxů `\levastrana` a `\pravastrana` a do registrů `\botversnumL` a `\botversnumP` uloží číslo posledního verše na potenciální levé, resp. pravé straně.

Po odkrojení obou potenciálních stran převezme řízení makro `\balancuj`, které by mělo fungovat zhruba takto: Pokud je číslo posledního verše v boxu `\levastrana` vyšší než číslo posledního verše v boxu `\pravastrana`, sníží velikost `\vsizeL` o výšku jednoho řádku, obnoví původní obsah boxu `\levybox` a vyvolá znovu makro `\odkrojlevou`. Pokud je číslo posledního verše v boxu `\levastrana` nižší než číslo posledního verše v boxu `\pravastrana`, sníží velikost `\vsizeP` o výšku jednoho řádku, obnoví původní obsah boxu `\pravybox` a vyvolá znovu makro `\odkrojpravou`. Pokud je číslo posledního verše v boxu `\levastrana` shodné s číslem posledního verše v boxu `\pravastrana`, vyvolá makro `\tiskni`, které v hlavním vertikálním módu pomocí `\unvbox` vloží obsah boxu `\levastrana` do vertikálního seznamu a pomocí `\penalty-10000` vyvolá výstupní rutinu, pak provede totéž s obsahem boxu `\pravastrana`.

Uvažujme však tento příklad: Verš číslo 10 je velice krátký a po rozložení levého textu do jednotlivých řádků leží na témže řádku začátek verše číslo 10 i začátek verše číslo 11; zároveň je tento řádek posledním řádkem v boxu `\levastrana`. Číslo posledního verše v boxu `\pravastrana` je 10. Makro `\balancuj` zmenší velikost `\vsizeL` o jeden řádek, takže po dalším zlomu, který provede makro `\odkrojlevou`, bude vlevo číslo posledního verše 9. Nyní tedy makro `\balancuj` zmenší velikost `\vsizeP`, a tak na právě sázené dvojstraně zůstane zbytečně nevyužito místo o velikosti jednoho řádku. Náš příklad by mohl být rozvinut ještě dále: Kratší by byly i jiné verše, takže by vlevo ležel začátek veršů číslo 8 a 9 na jednom řádku a vpravo by stejný jev nastal u veršů číslo 9 a 10. Za této situace by ještě více místa přišlo nazmar. Kdybychom příklad dovedli ad absurdum, nakonec by nám na právě sázené dvojstraně nezbyl vůbec žádný text. Z tohoto příkladu si vezmeme ponaučení, a do definice makra `\balancuj` zakomponujeme mechanismy, které podobným případům zabrání.

```

122 % test, zda se minule vešlo více vlevo než vpravo:
123 \newif\ifvetsiL
124 % test, zda se minule vešlo více vpravo než vlevo:
125 \newif\ifvetsiP
126 % test, zda se má skutečně balancovat:
127 \newif\ifbalancuj
128 \def\balancuj{\let\next=\tiskni
129   \ifdim\vsizeL>0pt \else
130     \ifnum\botversnumL>\botversnumP \balancujfalse \fi \fi
131   \ifdim\vsizeP>0pt \else
132     \ifnum\botversnumP>\botversnumL \balancujfalse \fi \fi
133   \ifbalancuj
134     \ifnum\botversnumL>\botversnumP \vetsiLtrue
135     \ifvetsiP
136       \advance\vsizeP by\baselineskip

```

```

137         \setbox\pravastrana=\vbox{}% uvolnění paměti
138         \setbox\pravybox=\box\zalohaprawybox
139         \def\next{\odkrojpravou \tiskni}%
140     \else
141         \advance\vsizel by-\baselineskip
142         \setbox\levastrana=\vbox{}% uvolnění paměti
143         \setbox\levybox=\box\zalohalevybox
144         \def\next{\odkrojlevou \balancuj}%
145     \fi
146 \else
147     \ifnum\botversnumL<\botversnumP \vetsiPtrue
148     \ifvetsiL
149         \advance\vsizel by\baselineskip
150         \setbox\levastrana=\vbox{}% uvolnění paměti
151         \setbox\levybox=\box\zalohalevybox
152         \def\next{\odkrojlevou \tiskni}%
153     \else
154         \advance\vsizel by-\baselineskip
155         \setbox\pravastrana=\vbox{}% uvolnění paměti
156         \setbox\pravybox=\box\zalohaprawybox
157         \def\next{\odkrojpravou \balancuj}%
158 \fi\fi\fi\fi \next}

```

Nyní nastala vhodná chvíle, abychom se začali zabývat tím, jaká opatření je třeba učinit v souvislosti s inserty.

Výchozí hodnota registrů `\vsizel` a `\vsizelP` by měla odpovídat maximální výšce textu, který se vejde na příslušnou stranu. Pokud bude text obsahovat poznámky a jiné inserty s koeficientem  $f > 0$ , tato maximální výška bude menší než `\vsizel`. Její hodnotu získáme tak, že kopii obsahu boxu `\levybox`, resp. `\pravybox` předáme standardním způsobem speciální výstupní rutině, která bude mít za úkol změřit výšku boxu 255 a pak zlikvidovat veškerý materiál z boxu 255 a ze všech boxů, ve kterých bude uložen obsah insertů. Zbytek materiálu, který zůstane ve vertikálním seznamu hlavního vertikálního módu, zlikviduje jiná speciální výstupní rutina. Jestliže budou tyto speciální výstupní rutiny vyvolány při kladném `\holdinginserts`, nebudeme se muset starat o to, které boxy jsou vyhrazeny pro inserty, neboť všechny inserty zůstanou v boxu 255.

Obsah boxů `\levastrana` a `\pravastrana` se má za pomoci makra `\tiskni` proměnit ve dvě strany uložené do souboru `dvi`. Kdybychom v makru `\tiskni` nevěnovali insertům náležitou péči, mohli bychom se dočkat nepříjemných překvapení. Algoritmus uzavření strany může totiž některé inserty vrátit z aktuální strany do přípravné oblasti; obsah těchto insertů by se pak buď vytiskl úplně jinde, než bychom chtěli, anebo by se vůbec nevytiskl. Proto pokud budou

některé inserty vráceny, vyvoláme speciální výstupní rutinu, a to při kladném `\holdinginserts`, takže všechny inserty zůstanou v boxu 255. Tato výstupní rutina přemístí inserty z boxu 255 do smlouveného boxu, odkud je pak vrátíme do boxu `\levybox`, resp. `\pravybox`. Zda algoritmus uzavření strany vrátil do přípravné oblasti nějaké inserty, se dozvíme dotazem na `\insertpenalties`. Tento registr má totiž dva různé významy: 1. V době činnosti algoritmu plnění strany se v registru `\insertpenalties` sčítají některé penalty související s inserty. 2. Algoritmus uzavření strany prostřednictvím registru `\insertpenalties` předá výstupní rutině informaci o počtu insertů vrácených do přípravné oblasti; jedná se o ty inserty, které byly v aktuální straně před místem zlomu a měly příznak „čekej“.

```

159 \newdimen\vyskatextu
160 \def\urcivyskuoutput{\setbox255=\vbox{\unvbox255}
161   \global\vyskatextu=\ht255
162   \global\output={\setbox0=\box255}}
163 \def\urcivyskuvlevo{\global\output={\urcivyskuoutput}
164   \global\holdinginserts=1
165   \unvcopy\levybox \break \vsizeL=\vyskatextu}
166 \def\urcivyskuvpravo{\global\output={\urcivyskuoutput}
167   \global\holdinginserts=1
168   \unvcopy\pravybox \break \vsizeP=\vyskatextu}

```

Při tvorbě kritického aparátu se obvykle preferuje sazba všech poznámek na příslušné straně do jediného odstavce, jak je to popsáno v Knuthově  $\TeX$ booku na stranách 398—400. Při tomto způsobu sazby poznámek se pracuje jen s odhadem výšky výsledného odstavce s poznámkami, takže se občas stává, že výška poznámek plus výška boxu 255 je o něco větší než `\vsize`, což se ve výstupní rutině projeví přetečeným `\vboxem` s tělem strany. Poněvadž při zjišťování maximální možné výšky textu každé strany voláme výstupní rutinu „nanečisto“, nabízí se vhodná příležitost popsaný nedostatek odstranit. V této výstupní rutině můžeme totiž zjistit badness boxu s tělem strany; pokud by se podle hodnoty badness ukázalo, že se jedná o přetečený box, dočasně bychom pro tuto stranu zmenšili `\vsize`. Pokusíme se tedy k příkazům `\urcivyskuvlevo` a `\urcivyskuvpravo` vytvořit alternativu s touto funkcí; bude-li si jí uživatel přát využít, pomocí příkazu `\let` změní implicitní význam zmíněných dvou příkazů za alternativní.

```

169 \newdimen\zalohavsize
170 \newcount\pagebadness
171 \def\urcivyskuoutputalt{\global\vsize=\zalohavsize
172   \setbox255=\vbox{\unvbox255}
173   \global\vyskatextu=\ht255
174   \setbox0=\pagebody \global\pagebadness=\badness
175   \global\output={\setbox0=\box255} \global\holdinginserts=1

```

```

176 \ifnum\insertpenalties>0 \null\break \fi}
177 \newdimen\VsizeL
178 \def\urcivyskuvlevoalt{\VsizeL=\vsize \let\strana=L
179 \loop
180 \global\output={\urcivyskuoutputalt}
181 \global\holdinginserts=0
182 \global\vsize=\VsizeL \unvcopy\levybox \break
183 \vsizeL=\vyskatextu
184 \ifdim\VsizeL>Opt \else \pagebadness=0 \fi
185 \ifnum\pagebadness=1000000 % overfull box
186 \advance\VsizeL by -0.5\baselineskip
187 \repeat}
188 \newdimen\VsizeP
189 \def\urcivyskuvpravoalt{\VsizeP=\vsize \let\strana=P
190 \loop
191 \global\output={\urcivyskuoutputalt}
192 \global\holdinginserts=0
193 \global\vsize=\VsizeP \unvcopy\pravybox \break
194 \vsizeP=\vyskatextu
195 \ifdim\VsizeP>Opt \else \pagebadness=0 \fi
196 \ifnum\pagebadness=1000000 % overfull box
197 \advance\VsizeP by -0.5\baselineskip
198 \repeat}

```

Speciální výstupní rutinu, kterou jsme nazvali `\urcivyskuoutputalt`, musíme vyvolat při nulovém `\holdinginserts`, poněvadž potřebujeme zjistit, zda odhad výšky odstavce s poznámkami byl tak nepřesný, že po zařazení tohoto odstavce do těla strany se tělo strany nevejde do `\vboxu` o výšce `\vsize`. Na řádce 174 předpokládáme, že makro `\pagebody` je synonymem pro vytvoření `\vboxu` obsahujícího tělo strany; hodnotu jeho `badness` si uschováme do registru `\pagebadness`. Někdy je při sestavování těla strany potřeba vědět, zda se jedná o levou, nebo pravou stranu; k tomu je určena řídicí sekvence `\strana`, která má při sestavování levé strany význam L a při sestavování pravé strany význam P. Povšimněme si, že registry `\VsizeL` a `\VsizeP` jsou určeny pro algoritmus stránkového zlomu a udávají cílovou výšku těla strany, zatímco registry `\vsizeL` a `\vsizeP` jsou určeny pro operaci `\vsplit` a udávají výšku textu, který má být zařazen do těla strany.

```

199 \newcount\zalohavbadness \newdimen\zalohavfuzz
200 \newcount\lastpenaltyL \newcount\lastpenaltyP
201 \newbox\boxins % sem budeme odchyťávat vrácené inserty
202 \def\tiskni{\vbadness=\zalohavbadness \vfuzz=\zalohavfuzz
203 \tisknilevou \tisknipravou}

```



```

204 \def\tisknilevou{\let\lastpagemarkL=\lastsplitmarkL
205   \setbox\zalohalevybox=\vbox{} % uvolnění paměti
206   \global\output={\tisknioutput} \global\holdinginserts=0
207   \let\strana=L \global\vsizel=\VsizeL
208   \unvbox\levastrana \lastpenaltyL=\lastpenalty \break
209   \doplnevybox
210   \ifvoid\boxins \else
211     \pridejinserty\levybox\lastpenaltyL \fi}
212 \def\tisknipravou{\let\lastpagemarkP=\lastsplitmarkP
213   \setbox\zalohapravybox=\vbox{} % uvolnění paměti
214   \global\output={\tisknioutput} \global\holdinginserts=0
215   \let\strana=P \global\vsizep=\VsizeP
216   \unvbox\pravastrana \lastpenaltyP=\lastpenalty \break
217   \doplpravybox
218   \ifvoid\boxins \else
219     \pridejinserty\pravybox\lastpenaltyP \fi}
220 \def\tisknioutput{\global\vsizel=\zalohavsize \usersoutput
221   \ifnum\insertpenalties>0 \saveins \fi}
222 \def\saveins{\global\holdinginserts=1
223   \global\output={\global\setbox\boxins=\vbox{
224     \unvbox255 \setbox0=\lastbox \unskip}
225     \global\output={\setbox0=\box255}}
226   \null\break}
227 \def\pridejinserty#1#2{\setbox#1=\vbox{\nobreak\vskip\topskip
228   \line{}}\nobreak\vskip-\topskip \unvbox\boxins
229   \ifnum #2=20000 \nobreak\fill\Supereject \fi \unvbox#1}}
230 \def\Supereject{\par\penalty20000\penalty-20000 }

```

V makru `\tiskni` je zapotřebí nejprve obnovit původní hodnotu registrů `\vbadness` a `\vfuzz`; při určování maximální výšky a při odkrajování jednotlivých stran budou totiž nastaveny na takovou hodnotu, aby byla potlačena varování `Underfull` a `Overfull` `\vbox`. Pak je vytištěna levá strana; hlavní činnost při tom vykoná výstupní rutina `\tisknioutput`, která je vyvolána příkazem `\break` poté, co je vypuštěn v hlavním vertikálním módu obsah boxu `\levastrana`. Pokud budou nějaké inserty vráceny z aktuální strany, příkaz `\saveins` se postará o jejich přesunutí do boxu `\boxins`, odkud poputují na začátek boxu `\levybox`, aby mohly být příště zařazeny na levou stranu. Na řádku 224 odebereme prázdný box, vytvořený o dva řádky níže, a mezeru z `\topskip`, takže v boxu `\boxins` zůstanou jen inserty. Poněvadž nemůžeme zjistit skutečnou hodnotu penalty v místě zlomu, nebude správně fungovat příkaz `\supereject`; uživatel by místo něj měl používat nově definovaný příkaz `\Supereject`. Podobně jako levá strana je vytištěna i pravá strana.

Cyklus, při kterém budou vytištěny jednotlivé dvojstrany celého dokumentu, se bude jmenovat `\synchronizuj`; zde je jeho kód:

```
231 \def\synchronizuj{\vbadness=10000 \vfuzz=\maxdimen
232   \urcivyskuvlevo \urcivyskuvpravo
233   \balancujtrue \vetsiLfalse \vetsiPfalse
234   \odkrojlevou \odkrojpravou \balancuj
235   \ifvoid\levybox
236     \ifvoid\pravybox \let\next=\relax
237     \else \let\next=\dokoncipravou \fi
238   \else
239     \ifvoid\pravybox \let\next=\dokoncilevou
240     \else \let\next=\synchronizuj \fi
241   \fi \next}
```

Makra `\dokoncilevou` a `\dokoncipravou` se uplatní tehdy, když se jeden z boxů `\levybox` nebo `\pravybox` úplně vyprázdnil a již není čím ho doplnit, zatímco ve druhém z těchto boxů je ještě nějaký materiál. Tato makra pak zajišťují, aby se tento zbylý materiál vysázal pouze na levé, resp. pravé strany a aby protilehlé strany zůstaly prázdné.

```
242 \def\dokoncilevou{\vbadness=10000 \vfuzz=\maxdimen
243   \urcivyskuvlevo \odkrojlevou
244   \vbadness=\zalohavbadness \vfuzz=\zalohavfuzz
245   \tisknilevou
246   \global\output={\usersoutput}
247   \let\strana=P \line{}\vfil\break % prázdná pravá strana
248   \ifvoid\levybox \let\next=\relax
249   \else \let\next=\dokoncilevou \fi
250   \next}
251 \def\dokoncipravou{\vbadness=10000 \vfuzz=\maxdimen
252   \urcivyskuvpravo \odkrojpravou
253   \vbadness=\zalohavbadness \vfuzz=\zalohavfuzz
254   \global\output={\usersoutput}
255   \let\strana=L \line{}\vfil\break % prázdná levá strana
256   \tisknipravou
257   \ifvoid\pravybox \let\next=\relax
258   \else \let\next=\dokoncipravou \fi
259   \next}
```

Zrcadlovou sazbu bude uživatel zahajovat pomocí makra `\zrcadli`. Jako první parametr uvede jméno souboru s levým textem, jako druhý parametr zapíše jméno souboru s pravým textem. Oba parametry budou mít jako separátor jednu mezeru. Makro `\zrcadli` se nejprve postará o to, aby zrcadlová sazba

začínala na prázdné levé (sudé) straně. Potom provede výchozí nastavení některých registrů a předá řízení makru `\synchronizuj`; tím se zahájí cyklus, při kterém se vytisknou jednotlivé strany dokumentu.

```
260 \def\zrcadli #1 #2 {\begingroup
261   \novalevastrana
262   \openin\levysoubor=#1 \openin\pravysoubor=#2
263   \zalohavsize=\vsize \VsizeL=\vsize \VsizeP=\vsize
264   \zalohavbadness=\vbadness \zalohavfuzz=\vfuzz
265   \maxdeadcycles=10000
266   \splittopskip=\topskip \splitmaxdepth=\maxdepth
267   \edef\usersoutput{\the\output}
268   \global\versnumL=0 \global\versnumP=0
269   \def\lastpagemarkL{0::} \def\lastpagemarkP{0::}
270   \global\prevdepthL=-1000pt \global\prevdepthP=-1000pt
271   \doplInlevybox \doplInpravybox
272   \synchronizuj
273   \global\output=\expandafter{\usersoutput}
274   \global\holdinginserts=0
275   \endgroup}
276 \def\novalevastrana{\supereject
277   \ifodd\pageno\line{ }\vfil\break\fi}
```

Pozorný čtenář si zajisté všiml, že na některých místech našeho kódu je globální přiřazení hodnot do některých registrů zdánlivě zbytečné. Toto globální přiřazování má ovšem svůj význam. Každá změna globální hodnoty registru na hodnotu lokální spotřebuje dvě slova z paměťového limitu „save size“. Proto střídáním globálního a lokálního přiřazování hodnoty jednomu a témuž registru můžeme tento limit snadno překročit. To je ostatně důvod, proč Knuth v plainu doporučuje užívat pomocné registry se sudými čísly a s číslem 255 pro lokální přiřazování a pomocné registry s lichými čísly (vyjma 255) pro globální přiřazování; toto doporučení bohužel nebývá občas uživateli respektováno.

V tuto chvíli je celý náš úkol vyřešen a nám už jen zbývá říci si pár slov o tom, jak budeme moci v roli uživatele pracovat se skupinou `maker`, která byla v tomto článku popsána. Veškerá tato makra uložíme třeba do souboru s názvem `zrcadlo.tex`.<sup>5</sup> Každý text, který budeme chtít užít při zrcadlové sazbě, budeme mít uložen v samostatném souboru a ve všech těchto souborech jednotlivé „verše“ opatříme smlouvenou synchronizační značkou — může to být buď přímo řídicí sekvence `\vers`, resp. `\Vers`, anebo jiná řídicí sekvence, kterou posléze nadefinujeme například tak, že nejdříve vytiskne konvenční označení verše, které bude mít uvedeno jako svůj parametr, a pak zavolá makro `\vers` či `\Vers`.

---

<sup>5</sup>Čtenář nemusí tato makra opisovat, neboť jsou ke stažení na autorových internetových stránkách <http://www.volny.cz/petr-brezina/>.

Celou sazbu budeme řídit z hlavního souboru, jehož základní struktura bude vypadat takto:

- načtení souboru maker `zrcadlo.tex` pomocí příkazu `\input`,
- různá nastavení a uživatelské definice,
- specifická nastavení pro levý a pravý text prostřednictvím řídicích sekvencí `\nastavlevy` a `\nastavpravy`,
- spuštění makra `\zrcadli`,
- ukončení činnosti  $\TeX$ u pomocí příkazu `\bye`.

Tuto strukturu si samozřejmě podle potřeby upravíme. Například můžeme zrcadlit více dokumentů za sebou.

Pokud bychom chtěli zrcadlit dvojici textů, v nichž by nebyly vyznačeny jednotlivé „verše“, nejjednodušší by bylo nastavit `\everypar={\vers}`. To je ovšem možné pouze za předpokladu, že každému odstavci z jednoho textu odpovídá právě jeden odstavec v textu druhém. Nebudou-li odstavce příliš dlouhé, výsledek bude uspokojivý. Nicméně nejlépe by bylo, aby oba texty byly rozděleny na úseky („verše“) zahrnující zhruba dva až tři řádky tištěného textu.

Při ručním doladování zrcadlové sazby můžeme v obou textech využít příkazu `\vadjust{\break}` pro vynucení stránkového zlomu na příslušném místě uvnitř odstavce.

Kromě třídy insertů `\footins`, kterou zřejmě použijeme pro kritický aparát a pro překladatelské poznámky, plain deklaruje ještě třídu insertů `\topins`, určenou pro plovoucí obrázky a tabulky. Na uživatelské úrovni se s touto třídou insertů pracuje pomocí řídicích sekvencí `\midinsert`, `\topinsert` a `\pageinsert`. Jedna z těchto řídicích sekvencí nebude v zrcadlové sazbě správně fungovat. Pilný čtenář si může rozmyslet, která to je a proč.

Ještě bychom chtěli upozornit na to, že inserty a značky typu `\write` by v zrcadlených textech měly být s předcházející linkou nebo boxem pevně spojeny, tak aby mezi oběma elementy nemohlo dojít ke stránkovému zlomu.

## Summary: Parallel typesetting

The article presents an efficient solution to the problem of typesetting two texts in parallel on facing pages in bilingual editions. The solution assumes that the two texts are saved separately in two files and that they are divided into small sections, just as the Bible is divided into verses. This division makes it possible to synchronize the texts automatically. Each of the two texts can have its own footnotes, illustrations and other insertions as if it were an ordinary document, but the texts are broken into individual pages simultaneously in such a way that each odd page contains the same sections as the corresponding even page. The presented macros are available on author's website <http://www.volny.cz/petr-brezina/>.

V článku *Zrcadlová sazba*, který by měl vyjít ve čtvrtém čísle Zpravodaje v roce 2008, jsem popsal, jak je možno postupovat při sazbě dvojjazyčných vydání. Když jsem se však rozhodl připravit latinsko-řecko-české vydání *Scipionova snu*, ocitl jsem se před trochu odlišným úkolem, který vyžadoval poněkud jiný přístup k řešení. Tři různé jazykové verze téhož díla jsem chtěl uspořádat tak, aby na každé dvojstraně byla příslušná část díla vždy ve všech třech verzích. V ideálním případě by latinský text zaujímal první dvě třetiny levé strany, český text první dvě třetiny pravé strany a řecký text by pokrýval vždy zbylou dolní třetinu strany pod latinským i českým textem. Tento ideál nemůže být ovšem ve skutečnosti dodržen, neboť tentýž významový úsek vyjádřený v různých jazycích má různou délku a cílem tohoto uspořádání je (na rozdíl od sazby dvojjazyčných vydání) vyplnit textem celou dvojstranu. A tak se může například stát, že latinský text zaplní více než dvě třetiny levé strany, český text méně než dvě třetiny pravé strany a zbývající místo na obou stranách bude vyplněno řeckým textem. K řešení podobných úkolů, kdy má být strana nebo v našem případě dvojstrana vyplněna několika různými texty, byly do  $\text{T}_{\text{E}}\text{X}$ u zabudovány mechanismy pro práci s datovým typem insert. Vzpomeňme si na poznámky pod čarou.

Základní idea řešení našeho úkolu je tato: Všechny tři jazykové verze budeme mít uloženy ve třech samostatných souborech. Text bude v těchto souborech rozčleněn pomocí smluvených řídicích sekvencí na relativně krátké úseky, které mohou začínat i uvnitř odstavce. Těmito úseky mohou být například jednotlivé věty. Důležité je, aby každému takovému úseku z jednoho souboru odpovídal v každém z obou ostatních souborů právě jeden obsahově shodný úsek. Pracovně budeme tyto úseky nazývat „verše“. Latinskou verzi budeme pokládat za základní text. Pro českou verzi zřídíme jednu třídu insertů a pro řeckou verzi druhou třídu insertů. Výšku strany nastavíme na dvojnásobek skutečné výšky strany, aby při stránkovém zlomu  $\text{T}_{\text{E}}\text{X}$  pracoval s celou dvojstranou, jako by se jednalo o jedinou stranu. Řídicí sekvenci, kterou jsme označili jednotlivé verše v latinském textu, definujeme tak, že vytvoří dva inserty, do nichž vloží příslušný český a řecký verš. Nyní se už  $\text{T}_{\text{E}}\text{X}$  sám postará o to, aby na každé dvojstraně byla odpovídající část textu latinsky, česky a řecky. Výstupní rutina bude do souboru dvi ukládat dvě strany: Nejdříve umístí box 255 čili latinský text do horní části strany, z boxu s řeckým textem odkrojí část odpovídající velikosti zbývajícího místa, vloží ji pod latinský text a celou stranu uloží do souboru dvi. Potom do horní části další strany dá box s českým textem, pod něj umístí box se zbytkem řeckého textu a celou stranu uloží do souboru dvi.

Nyní ještě zbývá vyřešit, jak budeme ze souborů s českým a řeckým textem odebírat jednotlivé verše. Tato činnost je ztížena skutečností, že verše mohou začínat a končit uvnitř odstavce. Zaměříme se na český text; s řeckým textem budeme zacházet obdobně. Řídící sekvenci označující jednotlivé verše v českém textu definujeme tak, aby do vertikálního seznamu pod příslušný řádek pomocí příkazu `\vadjust` vložila nějaký smluvený materiál, který nijak neovlivní sazbu. Může se jednat o box s celkovou výškou rovnou nule, jehož jeden rozměr bude mít magickou hodnotu, kterou v dalším procesu snadno rozpoznáme. Text budeme načítat po odstavcích do zvláštního boxu tak, aby celková výška načteného textu v tomto boxu byla stále o něco vyšší, než je výška strany; způsob tohoto načítání je podrobně popsán v článku *Zrcadlová sazba*, na nějž čtenáře odkazujeme. Dále budeme odzadu z tohoto boxu odebírat text po jednotlivých řádcích, dokud nenarazíme na onen magický box, signalizující konec jednoho verše a začátek dalšího. Všechny odebrané řádky patřící jednomu verši zapouzdříme do jednoho `\vboxu`, a tento `\vbox` uskladníme do jiného `\vboxu`. V odebírání řádků a jejich zapouzdřování po jednotlivých verších budeme pokračovat, dokud se onen box, do nějž jsme načetli text, nevyprázdní; zapouzdřené řádky budeme přidávat do onoho uskladňovacího `\vboxu`. Příkaz označující jednotlivé verše v latinském textu odebere vždy jedno pouzdro z uskladňovacího `\vboxu` a jeho obsah odpouzdří do insertu, o němž jsme hovořili v předchozím odstavci.

Je jasné, že jednotlivá pouzdra neobsahují pouze text, který patří jedinému verši, neboť verše nemusejí začínat na začátku řádku a končit na konci řádku; tyto přesahy nám ale vůbec nevadí. Konec verše může tedy být v jiném pouzdře než ostatní část verše; hledíme však na to, aby začátek verše byl pokud možno vždy v pouzdře náležejícím danému verši. Může se však stát, že některý verš je velmi krátký, a jeho začátek i konec leží na témže řádku spolu se začátkem dalšího verše. V takovém případě se v sazbě objeví dva magické boxy těsně za sebou a my ponecháme tento řádek verši, který je blíže začátku textu. Pokud by na některý verš nevyzbyl žádný řádek, vytvoříme pro něj prázdné pouzdro.

Dalším úskalím, na které si musíme dát pozor, je případ, kdy by nějaký verš byl rozdělen do dvou odstavců, mezi nimiž by došlo k předělu v načítání zdrojového textu. Bez náležité péče by se takový verš rozdělil na dvě části, z nichž by každá byla zapouzdřena zvlášť, což by pak vedlo k rozhození synchronizace textů.

Nyní bych rád předvedl praktickou ukázkou využití popsaného postupu při sazbě knihy *Scipionův sen*,<sup>1</sup> která obsahuje text ve třech jazycích — latinsky, řecky a česky; mimoto jsem mezi latinský a řecký text na levých stranách vložil aparát s textovými variantami. Dále nutno podotknout, že předmluva ke knize je jen dvojjazyčná (česko-latinská). Z tohoto důvodu a také proto, že některá

---

<sup>1</sup>Marcus Tullius Cicero: *Scipionův sen*. Petr Březina vlastním nákladem, Srní 2008. ISBN 978-80-254-0480-5. (Viz <http://www.volny.cz/petr-brezina/ss-cz.html>.)

z maker, která jsme již dříve definovali pro dvojjazyčnou sazbu, využijeme i při trojjazyčné sazbě, načteme nejprve kompletní makro pro dvojjazyčnou sazbu, jak jsme je popsali v již zmíněném článku *Zrcadlová sazba*:

```
1 \input zrcadlo
```

Podle maker pro načítání levého a pravého textu přidáme makra pro načítání třetího textu:

```
2 \newread\tretisoubor % zdrojový soubor se třetím textem
3 \newbox\tretibox % box, kam budeme načítat třetí text
4 \newdimen\prevdepthT % \prevdepth na konci boxu \tretibox
5 \let\nastavtreti=\relax % specifická nastavení pro třetí text
6 \def\doplntretibox{\let\next=\relax
7   \ifdim\ht\tretibox<1.25\vsiz
8     \ifeof\tretisoubor \else \nactitretitext
9       \let\next=\doplntretibox \fi\fi \next}
10 \def\nactitretitext{\let\soubo=\tretisoubor \kopiruj
11   \setbox\tretibox=\vbox{\break \unvbox\tretibox
12     \let\versnum=\versnumT \prevdepth=\prevdepthT
13     \nastavtreti \input\jobname.tmp
14     \global\prevdepthT=\prevdepth}%
15   \setbox0=\vsplit\tretibox to 0pt }
```

Pro pravý (tj. český) a třetí (tj. řecký) text zřídíme dvě nové třídy insertů. Zatímco třetí text může teoreticky zcela zaplnit obě dvě strany na dvojstraně, u pravého textu počítáme maximálně se zaplněním celé pravé strany. Protože kritický aparát má menší řádkování, a tedy rozhodí řádkový rejstřík, dáme mezeře oddělující tento aparát od třetího textu možnost mírně se stáhnout; poznamenejme, že `\bigskipamount` je při sazbě knihy roven `\baselineskip`, a nemá tedy žádnou pružnost.

```
16 \newinsert\pravyins
17 \newinsert\tretiins
18 \skip\pravyins=\bigskipamount
19 \count\pravyins=1000
20 \dimen\pravyins=\vsiz
21 \skip\tretiins=1\bigskipamount minus.25\bigskipamount
22 \count\tretiins=1000
23 \dimen\tretiins=2\vsiz
```

Pro uskladňovací boxy, do nichž si připravíme zapouzdřené verše, alokujeme příslušné registry:

```
24 \newbox\pripravenypravy
25 \newbox\pripravenytretri
```

Makro `\versL` bude označovat začátky jednotlivých veršů v levém (tj. latinském) textu. Poté co se přesvědčí, že je v odstavcovém módu, rozebere načtenou část pravého textu, jejíž výška odpovídá asi jedné a čtvrt výšky strany, na jednotlivé verše, které zapouzdří a uskladí. Pak je z uskládnovacího boxu vyjmuta jedno pouzdro a jeho obsah vložen do insertu. Pokud ale magická hodnota 13 sp signalizuje, že pouzdro je prázdné, insert vytvořen není. Obdobně je zpracován třetí text.

```

26 \def\versL{\leavevmode
27   \rozeberpravy
28   \global\setbox\pripravenypravy=\vbox{\unvbox\pripravenypravy
29     \global\setbox1=\lastbox}%
30   \ifvoid1 \else \ifdim\wd1=13sp \else
31     \insert\pravyins{\floatingpenalty=20000 \unvbox1}\fi\fi
32   \rozebertreti
33   \global\setbox\pripravenytreti=\vbox{\unvbox\pripravenytreti
34     \global\setbox1=\lastbox}%
35   \ifvoid1 \else \ifdim\wd1=13sp \else
36     \insert\tretiins{\floatingpenalty=20000 \unvbox1}\fi\fi}

```

Začátky jednotlivých veršů v pravém a třetím textu jsou označeny makry `\versP` a `\versT`, která do sazby vloží magický `\hbox`:

```

37 \def\versP{\leavevmode\vadjust{\hbox to12sp{}}}
38 \def\versT{\leavevmode\vadjust{\hbox to12sp{}}}

```

Teď přichází na řadu definice makra, které rozebere pravý, resp. třetí text na jednotlivé verše. Pokud je výška zapouzdřených a uskládněných veršů menší než 1,25 výšky strany, je načtena další část textu ze zdrojového souboru do boxu `\pravybox`, resp. `\tretibox` a následně odzadu rozebrána na jednotlivé verše, viz makro `\odebirej`. Případný rozdělený verš je v tuto chvíli sloučen do jednoho pouzdra; první zapouzdřená část rozděleného verše se nacházela na začátku uskládnovacího boxu, druhá zapouzdřená část na konci právě vytvořené skupiny pouzder, která je uložena v boxu 1. Potom je celá tato skupina pouzder přesunuta do uskládnovacího boxu.

```

39 \def\rozeberpravy{%
40   \ifdim\ht\pripravenypravy<.625\vsizel \dopl\pravybox \fi
41   \ifvoid\pravybox
42   \else
43     \setbox0=\box1 \setbox0=\box2 % vymažeme \box1 a \box2
44     \setbox\pravybox=\vbox{\unvbox\pravybox\odebirej}%
45     \ifrozdelenyvers
46       \global\rozdelenyversfalse
47     \global\setbox1=

```



```

48         \vbox{\unvbox1 \global\setbox3=\lastbox}%
49     \vemprvni\pripravenypravy
50     \global\setbox1=
51         \vbox{\unvbox1 \vbox{\unvbox5\unvbox3}}%
52     \fi
53     \global\setbox\pripravenypravy=
54     \vbox{\unvbox1\unvbox\pripravenypravy}%
55 \fi}
56 \def\rozebertreti{%
57     \ifdim\ht\pripravenytreti<0.625\vsizel\doplntretibox \fi
58     \ifvoid\tretibox
59     \else
60         \setbox0=\box1 \setbox0=\box2 % vymažeme \box1 a \box2
61         \setbox\tretibox=\vbox{\unvbox\tretibox\odebirej}%
62         \ifrozdelenyvers
63             \global\rozdelenyversfalse
64             \global\setbox1=
65                 \vbox{\unvbox1 \global\setbox3=\lastbox}%
66             \vemprvni\pripravenytreti
67             \global\setbox1=
68                 \vbox{\unvbox1 \vbox{\unvbox5\unvbox3}}%
69         \fi
70         \global\setbox\pripravenytreti=
71         \vbox{\unvbox1\unvbox\pripravenytreti}%
72     \fi}

```

Makro `\vemprvni` vyjme z daného boxu první `\vbox` a uloží ho do registru `\box5`:

```

73 \def\vemprvni#1{%
74     \setbox0=\vbox{\setbox2=\box0 % vymažeme \box0
75         \unvbox#1%
76         \loop
77             \global\setbox5=\box0
78             \setbox0=\lastbox
79         \ifvbox0
80             \global\setbox#1=\vbox{\box5\unvbox#1}%
81         \repeat}}

```

Jednotlivé elementy sazby jsou postupně odzadu odebírány pomocí makra `\odebirej`. Protože se mně nechtělo pracovat s vertikálními mezerami na rozhraní veršů, resp. insertů a protože jsem věděl, že v celém textu je jednotné řádkování, dovolil jsem si meziřádkové mezery ignorovat a místo toho jsem v makru

\odebirej nechal nastavit výšku všech řádků na hodnotu \baselineskip a jejich hloubku na nulu. Makro si všímá magických hodnot šířky boxů a pro jednotlivé verše vyváří pouzdra, která obsahují jen boxy a penalty. Magická hodnota 16 sp znamená, že výška a hloubka boxu nemá být měněna; tímto způsobem jsem v případě potřeby zajistil vertikální mezery.

```

82 \newif\ifrozdelenyvers
83 \def\odebirej{%
84   \count255=\lastpenalty \unpenalty
85   \advance\count255 by\lastpenalty \unpenalty
86   \setbox0=\lastbox
87   \unskip
88   \unskip
89   \ifhbox0
90     \let\next=\odebirej
91     \global\rozdelenyverstrue
92     \ifdim\wd0=12sp
93       \setbox0=\lastbox
94       \unskip
95       \unskip
96       \let\next=\uzavrivrs
97       \global\rozdelenyversfalse
98     \fi
99     \ifdim\wd0=12sp
100      \ifvoid2 \setbox2=\hbox to13sp{}\fi
101      \let\next=\vratvrs
102    \else
103      \ifdim\wd0=16sp
104        \else
105          \ht0=\baselineskip \dp0=0pt
106        \fi
107        \setbox2=\vbox{\box0\penalty\count255\unvbox2}%
108        \fi
109    \else
110      \ifrozdelenyvers
111        \let\next=\osetrirozdeleny
112      \else
113        \let\next=\relax
114      \fi
115    \fi
116  \next}
117 \def\uzavrivrs{%

```

```

118 \global\setbox1=\vbox{\unvbox1 \box2}%
119 \odebirej}
120 \def\vratvers{%
121 \global\setbox1=\vbox{\unvbox1 \box2}%
122 \nointerlineskip\box0 \odebirej}
123 \def\osetrirozdeleny{%
124 \global\setbox1=\vbox{\unvbox1 \box2}}

```

Pro aparát s textovými variantami zřídíme novou třídu insertů. Jakým způsobem jsou tyto varianty zapsány v souboru s latinským textem, se zde zabývat nebudeme. Důležitá je pro nás informace, že makro `\makeapparatusparagraph` vytvoří odstavec s textovými variantami, které se váží k latinskému textu na aktuální straně, resp. dvojstraně. Makro bylo inspirováno Knuthovým návodem na sazbu poznámek pod čarou do jediného odstavce.

```

125 \newinsert\apparatusins
126 \skip\apparatusins=\bigskipamount
127 \count\apparatusins=1000
128 \dimen\apparatusins=.75\vsiz

```

Výstupní rutina `\zrctriout` nejdříve nastaví `\vsiz` na výšku jedné strany; doposud byla totiž v tomto registru výška dvou stran. Toto nastavení je lokální a platí jen pro výstupní rutinu, takže po skončení činnosti výstupní rutiny se `\vsiz` vrátí ke dvojnásobné velikosti. Potom změříme, kolik místa zbude na levé straně pro třetí (tj. řecký) text, a takto velkou část třetího textu odkrojíme do boxu `\tretiboxL`. Pokud chceme mít jiné okraje na levých a jiné na pravých stranách, což bývá zvykem při sazbě knih, musíme náležitým způsobem definovat řídicí sekvence `\hoffsetL` a `\hoffsetP`, které předají náš požadavek registru `\hoffset`, který je zodpovědný za horizontální umístění sazby na straně. Výstupní rutina uloží do souboru dvi levou stranu, zvýší číslo strany o jedničku, uloží do souboru dvi pravou stranu a opět zvýší číslo strany o jedničku. Tím vlastně její úkol končí.

```

129 \newbox\tretiboxL
130 \def\zrctriout{%
131 \vsiz=.5\vsiz
132 \setbox255=\vbox{\unvbox255}
133 \dimen0=\vsiz
134 \advance\dimen0 by-\ht255
135 \ifvoid\apparatusins\else
136 \advance\dimen0 by-\skip\apparatusins
137 \advance\dimen0 by-\ht\apparatusins \fi
138 \advance\dimen0 by-\skip\tretiins
139 \advance\dimen0 by.25\skip\tretiins % = stažitelnost

```

```

140 \count255=\vbadness \vbadness=10000
141 \setbox\tretiboxL=\vsplit\tretiins to\dimen0
142 \vbadness=\count255
143 \ifx\hoffsetL\undefined \else \hoffset=\hoffsetL \fi
144 \shipout\vbox{\makeheadline\leftpagebody\makefootline}
145 \advancepageno
146 \ifx\hoffsetP\undefined \else \hoffset=\hoffsetP \fi
147 \shipout\vbox{\makeheadline\rightpagebody\makefootline}
148 \advancepageno
149 \ifnum\outputpenalty>-20000 \else\dosupereject\fi}

```

Makra `\leftpagebody` a `\rightpagebody` sestaví tělo levé a pravé strany. Na levou stranu přijde nejdříve levý text; pokud pro aktuální stranu existuje aparát s textovými variantami, je pod levý text vložena vertikální mezera a aparát, za nímž následuje pružná vertikální mezera, která má za úkol dorovnat řádkový rejstřík, který byl aparátem porušen, protože aparát je sázen menším písmem s menším řádkováním než okolní text; pak následuje další vertikální mezera, horizontální linka a třetí text. Tělo pravé strany je jednodušší: Za pravý text přijde vertikální mezera, horizontální linka a zbytek třetího textu; poněvadž by se však mohlo stát (zvláště na konci dokumentu), že se celý třetí text vešel vlevo, testujeme tuto skutečnost, aby se vpravo nevykreslila horizontální linka, pod níž by nebyl žádný text. Na tomto místě se sluší poznamenat, že z praktických důvodů bylo pro celou knihu nastaveno `\topskip=\baselineskip`, a jak za chvíli uvidíme, stejná hodnota je přiřazena registru `\splittopskip`. Zároveň připomínáme, že výška řádku byla v makru `\odebirej` nastavena na tutéž hodnotu. V důsledku těchto nastavení bude bez dalších úprav učaři prvního řádku třetího (tj. řeckého) textu na levých i pravých stranách shodně vzdáleno od linky, která ho odděluje od levého (tj. latinského), resp. pravého (tj. českého) textu.

```

150 \def\leftpagebody{\vbox to\vsizel{\boxmaxdepth\maxdepth
151 \dimen0=\dp255 \unvbox255 \unskip \kern-\dimen0
152 \ifvoid\apparatusins\else
153 \vskip\skip\apparatusins
154 \setbox0=\vbox{\makeapparatusparagraph \kern-\prevdepth}
155 \dimen0=\ht0
156 \divide\dimen0 by\baselineskip
157 \multiply\dimen0 by\baselineskip
158 \advance\dimen0 by-\ht0
159 \ifdim\dimen0<-.5\baselineskip
160 \advance\dimen0 by\baselineskip \fi
161 \nointerlineskip\box0
162 \vfil
163 \fi}

```

```

164 \vskip\skip\tretiins
165 \tretirule
166 \unvbox\tretiboxL}}
167 \def\rightpagebody{\vbox to\vsizel{\boxmaxdepth\maxdepth
168 \unvbox\pravysins
169 \ifvoid\tretiins
170 \else
171 \vskip\skip\tretiins
172 \tretirule
173 \unvbox\tretiins
174 \fi
175 \vfil}}
176 \def\tretirule{\kern-5pt \hrule \kern 4.6pt}

```

Ted už nám zbývá jen definovat uživatelské makro `\zrcetri` na zahájení paralelní sazby tří textů. Makro má tři parametry, za něž je třeba dosadit jména tří souborů obsahujících texty, které chceme vysázet. Jak jsme již dříve řekli, texty musejí být pomocí řídicích sekvencí `\versL`, `\versP` a `\versT` rozděleny na jednotlivé verše.

```

177 \def\zrcadlitri #1 #2 #3 {\begingroup
178 \openin\pravysoubor=#2 \openin\tretisoubor=#3
179 \ifdim\topskip=\baselineskip \else
180 \errmessage{Velikost \string\topskip\space není rovna
181 \string\baselineskip}\fi
182 \topskip=1\topskip % odstraníme případnou pružnost
183 \splittopskip=\topskip \splitmaxdepth=\maxdepth
184 \global\rozdelenyversfalse
185 \vsizel=2\vsizel
186 \output={\zrcetriout}
187 \nastavlevy
188 \input #1
189 \vfil\supereject
190 \endgroup}

```

Podívejme se nyní trochu blíže na to, jak  $\text{T}_{\text{E}}\text{X}$  postupuje při zařazování jednotlivých veršů na aktuální dvojstranu. Nejdříve je zařazen první řádek latinského verše; za ním se v sazbě objeví insert s odpovídajícím českým veršem a  $\text{T}_{\text{E}}\text{X}$  se pokusí umístit celý tento český verš na aktuální dvojstranu. Potom přijde na řadu insert s odpovídajícím řeckým veršem, který je opět pokud možno celý vložen na aktuální dvojstranu. Nakonec jsou na aktuální dvojstranu přidány zbývající řádky latinského verše, pokud tam pro ně ještě je místo. Tento postup zařazování veršů na aktuální dvojstranu se občas u posledního verše na dvojstraně projevil negativně, neboť poslední český verš zůstává na dvojstraně zpra-

vidla celý, zatímco z jeho latinského protějšku může zbyť na příslušné dvojstraně třeba jen jediný řádek a jeho zbytek je pak odsunut na začátek další dvojstrany. Tento nepoměr lze zmírnit rozdělením výchozích textů na co možná nejkratší verše. Dvou- až třířádkové verše jsou obvykle dostačující. Nicméně při ručním dolaďování sazby můžeme v případě výskytu problematického verše tento verš ad hoc rozdělit na dvě kratší části vložním příkazů `\versL`, `\versP` a `\versT` na potřebné místo ve všech třech zdrojových souborech.

Veškerý kód prezentovaný v tomto článku nalezne čtenář v souboru nazvaném `zrcetri.tex` na mých webových stránkách <http://www.volny.cz/petr-brezina/>.

## Summary: Typesetting of a trilingual book

$\text{T}_{\text{E}}\text{X}$  has an insertion mechanism that makes it possible to handle several texts simultaneously. It can be used in preparation of multilingual books. In this article, the author describes how he has typeset a Latin-Greek-Czech edition of *The Dream of Scipio* where each double page contains the text simultaneously in the three languages. The described macros are available on author's home page <http://www.volny.cz/petr-brezina/>.

Po čtyřech letech se uskutečnilo další setkání českých a slovenských T<sub>E</sub>Xistů. Tentokrát pod názvem T<sub>E</sub>Xperience v podzimně mlhavých Hostýnských vrších. Pořadatelství se ujal Pavel Stríž s rodinou a Zdeněk Wagner. Zhostili se ho s takovou vervou, že mou nabízenou pomoc nevyužili více, než rolí konferenčního zpravodaje, kterou nadneseně (alespoň doufám) nazvali historikem. Pokusím se tedy tímto stručně T<sub>E</sub>Xperience 2008, za který patří Pavlovi se Zdeňkem velký dík, přiblížit.

- Setkání se konalo ve východní části moravských Beskyd, v centru Hostýnských vrchů, na stráni nad v 16. století rumunskými valachy kolonizovanou obcí Rusava, v chatě Jestřabí ([www.jestrabici.cz](http://www.jestrabici.cz)).
- Přivítání bylo překvapivé a tradiční zároveň: chlebem, solí a – slivovicí. Jednak ořechovou a druhak meruňkovou. Tento úvod hned na začátku navodil příjemnou uvolněnou atmosféru, která vydržela až do konce setkání.
- Pro účastníky bylo přichystáno konferenční číslo Zpravodaje 3/2008 – poprvé v jeho historii s barevnou obálkou. Příspěvků bylo tolik, že se do jednoho čísla nevešly, takže zbytek vychází v tomto čísle 4/2008.
- Kromě něj každý účastník dostal dvě plné igelitky konferenčních materiálů. Mezi nimi se vyjímalala kniha pohádek Zdeňka Wagnera *Sedmero dračích srdcí*, kniha cestovatele Milana Štourače *Ze zápisů dobrodružáčka* a placatice domácí meruňkovice s ruční výšivkou T<sub>E</sub>Xperience 2008 z dílny rodičů Pavla Stríže.
- O žaludek bylo řádně postaráno výbornou domácí krmí majitelů chaty, a to včetně druhých večeří. Minerálky, pivo, víno a slivovice byly účastníkům volně k dispozici v dostatečném množství.
- Přednesené příspěvky nemá cenu blíže představovat, jejich psané verze jsou ve zpravodajích. Za zmínku stojí výborný třídílný a ve Zpravodaji nepublikovaný tutoriál Petra Olšáka o makrojazyku T<sub>E</sub>Xu, ve kterém byly zajímavé již názvy jednotlivých částí: lednice, panenka, blbost, sekvence, závorky, citace.
- Na programu byly také dvě panelové diskuse:
  - **LTUG** Tuto diskusi uvedl předseda **LTUGu** Jaromír Kuben smutnou zprávou, že žádný z následujících úkolů z výroční schůze **LTUGu** konané v loňském prosinci nebyl splněn:
    - \* administrativní rozhraní k členské databázi pro sekretářku Sdružení (generování šitků pro rozesílání tiskovin členům, generování faktur kolektivním členům),



Vít Zýka na TeXperience 2008. Foto Zdeněk Wagner



- \* úplné rozhraní členské databáze pro sekretářku (ta, jako jediná placená síla  $\zeta$ TUGu, svou práci kvůli absenci rozhraní nemůže plně vykonávat),
- \* opravu dat v členské databázi,
- \* zavedení redakčního systému pro snadnou správu webu `cstug.cz`,
- \* robot pro automatické upozorňování na placení příspěvků.

Jde o dlouhodobé nedostatky způsobující problémy členům i výboru. Důsledkem je rozčarování a špatná informovanost členů i jejich úbytek. Důvodem tohoto stavu je, že  $\zeta$ TUG odmítl drahé komerční řešení administrace a jediný dobrovolník, Miroslav Fridrich (zprostředkováno Zdeňkem Wagnerem), který se k této práci uvolil, ji pro zaneprázdněnost neprovedl.

Dobrá zpráva je, že se našlo řešení: Využijeme nabídku Marcela Takáče na implementování nové členské databáze (v té staré potřebné funkce chybí a kvůli mnoha záplatám od několika správců, je kód již takřka needitovatelný) spolu s uživatelským rozhraním. Specifikaci této aplikace slíbil napsat pan Wagner do 12. 10. Na základě ní sepíšeme dohodu o provedení práce s panem Takáčem. Ve smlouvě počítáme s uvedením termínu dokončení pravděpodobně k 1. 2. 2009. Cena bude kompromisem mezi cenou komerční firmy a dobrovolnou prací, řádově v desítkách Kč.

Než bude nový systém nasazen, prosíme všechny členy, aby měli s administrativou ještě chvíli strpení. Pokud jste dosud neuhradili letošní příspěvky, zůstaňte  $\zeta$ TUGu věrni a, prosím, uhradte je. Připomínám, že členstvím přímo či nepřímo přispíváte na projekty, které při svém  $\text{T}_{\text{E}}\text{X}$ ování využíváte (distribuce  $\text{T}_{\text{E}}\text{X}$ Live, písma  $\text{T}_{\text{E}}\text{X}$  Gyre, METAPOST knihovna,  $\text{LuaT}_{\text{E}}\text{X}$ ), a díky kterým  $\text{T}_{\text{E}}\text{X}$  může zůstat konkurenceschopný jiným sázecím systémům.

Problém vygenerování faktur kolektivním členům, který je akutní, vyřeší pan Wagner.

–  **$\text{T}_{\text{E}}\text{X}$ Live 2008 (TL08)** V tomto druhém panelovém bloku byly zprávy mnohem veselejší:

- \* 8GB DVD (i s archivem CTAN) je v tisku a bude distribuováno do konce roku platícím členům spolu se Zpravodajem 4/2008. Pro něj je dostatek materiálu z  $\text{T}_{\text{E}}\text{X}$ perience, který se nevešel do sborníkového vydání 3/2008.
- \* V TL08 je nově implementován balíčkovací systém s možným síťovým upgradem jednotlivých balíčků v rámci dané verze  $\text{T}_{\text{E}}\text{X}$ Live. Též je již možná úplná síťová instalace bez DVD.
- \* Po dvou letech reagoval Johannes Braams a začlenil CSBabel Petra Tesaříka do oficiální distribuce. Balíček Zdeňka Wagnera se tak stává pro běžného uživatele zbytečným, vše pro české a slovenské

psaní v L<sup>A</sup>T<sub>E</sub>Xu je ve standardní distribuci. Dík za vykorespondování patří Petru Sojkovi.

- \* Ján Buša aktualizoval českou/slovenskou dokumentaci TL08.
- \* Bylo konstatováno, že TL08 lze provozovat paralelně se starými instalacemi T<sub>E</sub>Xu. Jen je dobré v době instalace TL08 vyřadit z proměnné PATH všechny adresáře s binárkami jiných TL.
- \* Do distribuce se vrátil program vlna Petra Olšáka, ale ne CSIndex, protože nebyl nikdo, kdo by pro něj vytvořil autoconf skript. Pokud se někdo najde, nic jeho zařazení nebrání. Již v současné době je však na TL08 balík XINDY s českou podporou, který je nadmnožinou CSIndexu se stejným uživatelským rozhraním. Petr Sojka se pokusí u autora této české lokalizace zajistit článek do Zpravodaje.
- Protože nejen T<sub>E</sub>Xem živ je člověk, vyrazili jsme na celé odpoledne přes podzimní mlhou zahalené Hostýnské vrchy na jedno z našich nejnámějších poutních míst – vrch Hostýn s chrámem Nanebevzetí Panny Marie pyšníci se papežským titulem Menší baziliky (Basilica minor).
- Organizátoři se postarali i o kulturní program. Zdeněk Wagner a Milan Štourač měli autorské čtení a autogramiádu, druhý jmenovaný i fotografické pásmo o svých putováních Albánií, Rumunskem a Moldavskem. Též jsme shlédli dva taneční výstupy dámské skupiny Kirké, jeden španělský, druhý orientální na směs melodií přelévající se někde mezi popem a technem (bohužel ne T<sub>E</sub>Xnem).
- V nabitém programu došlo k ránu i na kytary a zpěv. Postaral se o to Jaroslav Hajtmar a Michal Polášek.

Z výše uvedeného vysvítá, že setkání bylo nadmíru povedené. Proslýchá se, že organizátoři chtějí z T<sub>E</sub>Xperience udělat pravidelnou roční akci a druhý ročník by měl proběhnout již na jaře 2009!

*Vít Zýka* vit.zyka@seznam.cz

Přináším několik heslovitých postřehů z druhého setkání uživatelů ConT<sub>E</sub>Xtu ve slovinšské Bohini. Z Česka se zúčastnil Karel Píška (s příspěvkem *Making Type 1 and OpenType fonts with MetaType1 and FontForge*), Jano Kula a Vít Zýka. Setkání bylo příjemné, neformální a proběhlo hladce.

- Hlavní organizátorkou byla Mojca Miklavec. Vystoupilo 17 řečníků s 26 řádnými příspěvky před téměř 30 účastníky. Hlavními postavami byli, kromě organizátorky, Taco Hoekwater, Hans Hagen a Arthur Reutenauer.
- Ústředním tématem setkání byl LuaT<sub>E</sub>X: instalace, ConT<sub>E</sub>Xtování v něm, nástroje a samozřejmě ukázky dosud nepředstavitelných triků z klávesnice Hanse Hagen a Taco Hoekwatera.
- Diskusní panel byl věnován dokumentaci. Dobrá zpráva je, že Taco dává do kupy ucelený ConT<sub>E</sub>Xtový manuál dosud roztroušený do mnoha dokumentů. Horší zpráva je, že dle mého názoru mnohem potřebnější referenční manuál (typu tex-show) nemá u hlavních vývojářů vysokou prioritu.
- Jano Kula neváhal do svého kufru naskládat 70 plechovkových plzní pro své ConT<sub>E</sub>Xtové přátele, ačkoliv cestoval na konferenci vlakem a autobusem.
- V jedné noční či spíše ranní pivní diskusi vzniklo logo ConT<sub>E</sub>Xtu, viz obr. 1, umístěné hned na stránky ConT<sub>E</sub>Xtové zahrady:  
<http://contextgarden.net>.



Obrázek 1: Návrh loga pro ConT<sub>E</sub>Xt.

- Součástí setkání byly dva semináře v Technickém muzeu ve městě Bystra. Při jednom jsme si mohli z cínu odlít raznici s logem T<sub>E</sub>Xu a iniciálové kuželky, viz obr. 2. Při druhém jsme si svázali knížечku o novém potenciálu písma Punk Donalda Knutha v LuaT<sub>E</sub>Xu.



Obrázek 2: Během workshopů si každý mohl odlít raznici a iniciálové kuželky.

- Abychom jen neseděli v hotelu, navštívili jsme přírodní raritu Slovinska – Škocjanske jame. Jde o obrovské podzemní prostory v krasové oblasti jménem Kras (odtud pochází tento mezinárodní termín) s ponornou řekou Reka.
- Program a abstrakty příspěvků jsou k nalezení na adrese <http://meeting.contextgarden.net/2008/programme.shtml>, k některým i prezentace a zdrojové kódy.

*Vít Zýka* [vít.zyka@seznam.cz](mailto:vít.zyka@seznam.cz)

Vážené kolegyně, vážení kolegové,

pořadatelé, Československé sdružení uživatelů T<sub>E</sub>Xu a Fakulta managementu a ekonomiky Univerzity Tomáše Bati ve Zlíně, vás zvou na 2. ročník odborné konference T<sub>E</sub>Xperience.

Ta se uskuteční v prodlouženém víkendu, **21. – 24. května 2009** v rekreačním středisku Královec, [www.kralovec.cz](http://www.kralovec.cz), Valašské Klobouky, okres Zlín, u hranic se Slovenskou republikou.

*Motto akce:* „Zažijme extázi T<sub>E</sub>Xu: T<sub>E</sub>Xem totiž řešíme vše, věci možné i nemožné!“

Přednášky předběžně přislíbili:

- Karel Horák: Práce s postscriptem při práci s grafikou.
- Karel Horák: Cvičení s MetaType1.
- Karel Píška: Fonty a jak s nimi pracují (plain)<sub>T<sub>E</sub>X</sub>, L<sub>A</sub><sub>T<sub>E</sub>X</sub>, pdf<sub>T<sub>E</sub>X</sub>, X<sub>Y</sub><sub>T<sub>E</sub>X</sub> a Lua<sub>T<sub>E</sub>X</sub>.
- Michal Růžička: Použití SVN pro verzování T<sub>E</sub>Xových dokumentů.
- Petr Březina: Využití T<sub>E</sub>Xu při sazbě zrcadlových vydání.
- Vít Zýka: CON<sub>T<sub>E</sub>X</sub>T prakticky.
- Jiří Rybička: Jak na závěrečné práce na vysoké škole.
- Pavel Stříž: Příprava trojboje šablon na vysokoškolské kvalifikační práce.
- Student Petr Nevřiva: Balíček PSTricks PST-GEO.
- Studentka Martina Kudláčková: Výroba výukových flash animací.
- Tomáš Hála: Výroba cenin a jiných lahůdek.
- Tomáš Hála: Povídání o papíru.

*Příspěvky* dalších přednášejících budou včas upřesněny v Programu konference.

*Poustržky* libovolné velikosti lze dovézt.

*Studenti* na konferenci vítáni. *Panelová diskuze* bude.

*Výzva:* dovezte své i cizí počiny (knihy, skripta, články, postery, ...) dělané pomocí T<sub>E</sub>Xu a spřátelených programů, uděláme opět hromadnou výstavku.

*Sponzoři* vítáni. Kontaktujte prosím přímo organizátory.

**Termín** vyplnění přihlášky na [cstug.cz](http://cstug.cz) a uhrazení poplatku na bankovní účet je jednotný: **30. dubna 2009**; termín dodání plného textu v šabloně Zpravodaje na adresu [bulletin@cstug.cz](mailto:bulletin@cstug.cz) je **31. května 2009**. Plné texty budou zveřejněny ve Zpravodajích 2/2009 a 3/2009.

- Pozdní uhrazení by snad bylo možné řešit individuálně nebo v hotovosti na místě.
- Pozdní přihlášení je snad možné řešit s organizátory na základě aktuální kapacity.
- Pozdní odeslání příspěvku znamená, že bude zveřejněn až ve Zpravodaji 3/2009.

*Cena:* člen  $\zeta$ TUGu 2000 Kč (student 1500 Kč), nečlen 2500 Kč (student 2000 Kč), doprovodná osoba 1580 Kč.

*Stravování a občerstvení.* Je možné v místní restauraci a baru.

*Slevy.* Při pozdním příjezdu nebo předčasném odjezdu nejsme schopni nabídnout slevu.

*Technika:* dataprojektor, Internet snad také bude.

*Doprava:* autem na středisko, parkovat na místě lze. Vlakem nebo autobusem (Valašské Klobouky). Plus 3,5 km pěšky na středisko nebo po zavolání vás vyzvedneme autem.

*Ubytování:* převážně dvoulůžkové v bungalovech a chatách.

*Doprovodné osoby:* ano, jsou možné, i dospělé i děti, i psi i kočky.

*Toalety a sprchy* jsou v každé ubyt. jednotce.

*Počasí* je domluveno. Teplé oblečení s sebou!

*Potenciál výletů:* Valašské Klobouky (4 km), Návojná s klubovnou Skalních orlů (historie skautingu; 4 km), židovský hřbitov a hrad Brumov (10 km), Rozhledna na Královci (1 km), . . .

Táboráček, posezení u ohně, kytarové tóniny a sběr hub, vše je možné.

*Organizátoři akce:*

Zdeněk Wagner <bulletin@csstug.cz>,

Pavel Stríž <striz@fame.utb.cz>, telefon: +420 603 380 402,

na vlastní akci také +420 724 009 340

(s dopravou a nouzovými stavy rád pomůže táta, Milan Stríž).



**KRÁLOVEC**

---

---

# EuroT<sub>E</sub>X 2009 & 3rd ConT<sub>E</sub>Xt Meeting

---

In 2009, the EuroT<sub>E</sub>X and ConT<sub>E</sub>Xt meetings will both be held together at the same time and place in the Netherlands, on August 24 through 28, and the conference will focus on educational uses of T<sub>E</sub>X.

The organisation of the conference is a joint effort of the NTG and the ConT<sub>E</sub>Xt task force. The conference is made possible by the Netherlands Defence Academy (NLDA) that graciously invited us to their facilities in The Hague.

These pages will be updated as more information becomes available. If you wish to contact the organisers immediately, you can send an email message to [eurotex@ntg.nl](mailto:eurotex@ntg.nl).

## Location

The conference location will be the Netherlands Defence College (NDC) in The Hague, one of the educational sites of the Netherlands Defence Academy (NLDA).

NLDA mission: initial officer training and scientific education for Netherlands Army, Navy and Air Force. Three locations: Breda (Royal Military Academy), Den Helder (Royal Netherlands Naval College) and The Hague (Netherlands Defence College).

The NDC location has numerous lecture halls and classrooms of varying size. The main conference room, the Bolman auditorium, has 200 fixed seats and all the modern audio-visual equipment and interpreter booths. Parallel sessions and tutorials will be held in one or two of the smaller classrooms.

## Accommodation

Meals are being served at the NDC local restaurant.

The NDC has an on-site hotel with more than two hundred individual single-person rooms. About 80 of those are reserved for our use. Each of the spacious rooms has a private bathroom with shower, a desk, cable television, and a lan connection to the internet. The on-site hotel is available at *no extra charge* and comes highly recommended.

However, if for some reason you prefer not to stay on-site (and note that there are *no* double rooms), it is best to look for hotels in Delft, as the Delft city center

is much closer by than The Hague. There are many on-line booking sites, just search for “Hotel” and “Delft”. For instance, you could use [Booking.com](http://Booking.com).

The Campanile and the China Hotel are within easy walking distance, but if you like a cozy atmosphere, you should go for one of the many hotels in the city centre.

## Call for papers

As usual, proposals for presentations and workshops are welcomed on just about any topic of interest to T<sub>E</sub>X users, but the conference focus will be on **Educational uses of T<sub>E</sub>X** such as manuals, courseware and college presentations, so we especially welcome proposals on subjects in those fields.

The language of the conference is English. Please send abstracts and proposals in plain text or T<sub>E</sub>X format to the conference committee [eurotex@ntg.nl](mailto:eurotex@ntg.nl)

### Timeline

Paper submissions due (abstracts)	Thu 30 April 2009
Notifications	Fri 15 May 2009
Conference date	24–28 August 2009
Final papers due	Mon 14 September 2009
Proceedings published	Mon 30 November 2009

## Registration

The conference fee is all-in: everything except for the full day excursion and following dinner is included. All the conference materials, meals, and on-site hotel accommodation from the evening of the 23rd up to the morning of 29th will be covered by a single sum. The registration fee is

- **350 euro** for Local T<sub>E</sub>X User Group (LUG) members
- **400 euro** for all others (including a one year membership of a LUG of your choice).

There is a **100 euro** discount for early bird registration (registration and payment received before Feb 1, 2009).

Attending the full-day excursion and the following conference dinner on Wednesday the 26th will add an additional **75 euro**.

*<http://www.ntg.nl/EuroTeX2009/>*



---

---

# TUG 2009

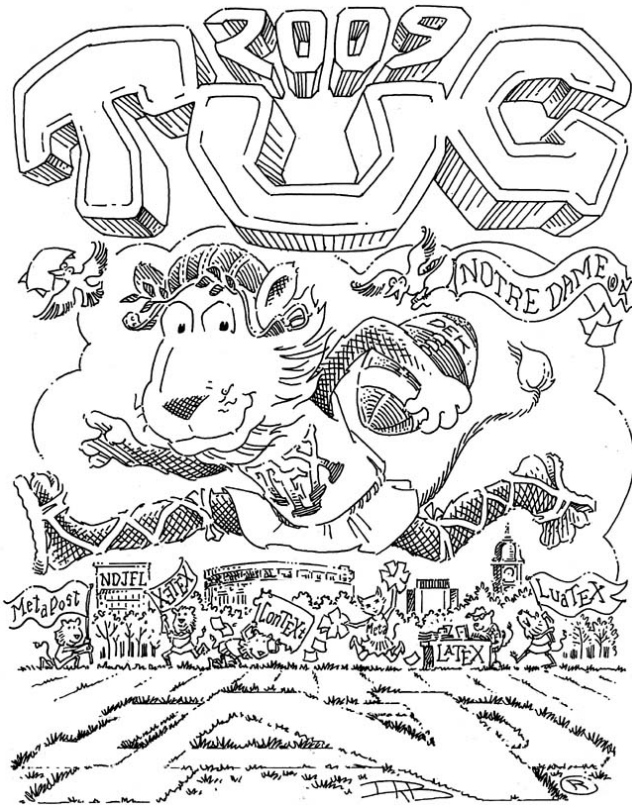
---

TUG 2009 will be held July 28–31, 2009, at the University of Notre Dame, in Notre Dame, Indiana, USA. The nearest airport is South Bend, Indiana (SBN). The location is about 90 miles southeast of Chicago.

The principal local organizer is Martha Kummerer, of the Notre Dame Journal of Formal Logic, which is sponsoring the conference.

Email contact: [tug2009@tug.org](mailto:tug2009@tug.org).

More when the time draws closer!



<http://tug.org/tug2009/>

**Zpravodaj Československého sdružení uživatelů T<sub>E</sub>Xu**  
ISSN 1211-6661 (tištěná verze), ISSN 1213-8185 (online verze)

Vydalo: Československé sdružení uživatelů T<sub>E</sub>Xu  
vlastním nákladem jako interní publikaci  
Obálka: Antonín Strejc  
Počet výtisků: 600  
Uzávěrka: 15. 11. 2008  
Odpovědný redaktor: Zdeněk Wagner  
Redakční rada: Pavel Stríž, David Catto a výbor CSTUGu  
Tisk a distribuce: KONVOJ, spol. s r. o., Berkova 22, 612 00 Brno,  
tel. +420 549 240 233  
Adresa: ČSTUG, c/o FEL ČVUT, Technická 2, 166 27 Praha 6  
Tel: +420 224 353 611  
Fax: +420 233 332 938  
Email: [cstug@cstug.cz](mailto:cstug@cstug.cz)

Zřízené poštovní aliasy sdružení ČSTUG:

[bulletin@cstug.cz](mailto:bulletin@cstug.cz), [zpravodaj@cstug.cz](mailto:zpravodaj@cstug.cz)  
korespondence ohledně Zpravodaje sdružení

[board@cstug.cz](mailto:board@cstug.cz)  
korespondence členům výboru

[cstug@cstug.cz](mailto:cstug@cstug.cz), [president@cstug.cz](mailto:president@cstug.cz)  
korespondence předsedovi sdružení

[gacstug@cstug.cz](mailto:gacstug@cstug.cz)  
grantová agentura ČSTUGu

[secretary@cstug.cz](mailto:secretary@cstug.cz), [orders@cstug.cz](mailto:orders@cstug.cz)  
korespondence administrativní síle sdružení, objednávky CD-ROM

[cstug-members@cstug.cz](mailto:cstug-members@cstug.cz)  
korespondence členům sdružení

[cstug-faq@cstug.cz](mailto:cstug-faq@cstug.cz)  
řešené otázky s odpovědmi navrhované k zařazení do dokumentu ČSFAQ

[bookorders@cstug.cz](mailto:bookorders@cstug.cz)  
objednávky tištěné T<sub>E</sub>Xové literatury na dobírku

ftp server sdružení:  
<ftp://ftp.cstug.cz>

www server sdružení:  
<http://www.cstug.cz>



Chata Jestřabí, místo konání T<sub>E</sub>Xperience 2008. Na zadní straně jsou další snímky z konference. Foto Zdeněk Wagner

