

□ □ □ □ □ □ □ **ÚVOD DO TYPOGRAFIE**
□ □ □ □ □ □ □ **A POČÍTAČOVÉ GRAFIKY**
□ □ □ □ □ □ □ **Vladimír P. Polách**

Univerzita Palackého v Olomouci □ □ □ □ □ □ □
2008 □ □ □ □ □ □ □

TENTO PROJEKT JE SPOLUFINANCOVÁN EVROPSKÝM SOCIÁLNÍM
FONDEM A STÁTNÍM ROZPOČTEM ČESKÉ REPUBLIKY

1. vydání

© Vladimír P. Polách, 2008

ISBN 978-80-244-2063-9

OBSAH

1	DTP, pre-press a ti jiní...	5
2	Softwarové vybavení	7
2.1	Textové editory	7
2.2	Textové procesory	10
2.3	DTP programy	11
2.4	Ostatní software	12
3	Grafika	14
3.1	Vektorová grafika	14
3.1.1	Výhody vektorové grafiky	14
3.1.2	Softwarové vybavení	15
3.1.3	Nejrozšířenější formáty	16
3.1.4	PostScript a PDF	17
3.1.5	Trasování obrázků	18
3.2	Bitmapová grafika	18
3.2.1	Výhody (a nevýhody) bitmapové grafiky	19
3.2.2	Softwarové vybavení	20
3.2.3	Nejrozšířenější formáty	21
3.2.4	Problematika komprese	22
3.2.5	Rozlišení obrazu, DPI a LPI	23
3.2.6	Screen capture	25
4	Barevné modely a práce s barvami	27
4.1	Barevné modely	27
4.1.1	Model RGB	27
4.1.2	Model CMYK	28
4.1.3	Jiné modely pro práci s barvami	29
4.2	Barevná hloubka	30
4.3	Gamut barev	31
4.4	Přímé barvy, duplex	31
5	Písmo	33
5.1	Písmo jako předmět typografie	33
5.1.1	Řez a rodina	33
5.1.2	Verzátky, kapitálky, minusky...	34
5.1.3	Kresba a konstrukce písma	34
5.1.4	Velikost písma	35
5.1.4.1	Didotův měrný systém	35

5.1.4.2	Měrný systém point/pica	35
5.1.4.3	Kuželka & Čtverčík	36
5.1.4.4	Tradiční udávání velikosti písma	36
5.1.5	Kerning a prostrkání	37
5.1.6	Klasifikace písem	38
5.1.6.1	Solperova klasifikace	38
5.1.6.2	Zjednodušená klasifikace podle druhů písem	38
5.1.6.3	Proporcionální a neproporcionální písma	39
5.2	Písmo jako předmět počítačové techniky	39
5.2.2	Formáty písem (fontů)	40
5.2.2.1	TrueType	40
5.2.2.2	Type 1	40
5.2.2.3	OpenType	41
5.2.3	Práce s písmem	41
5.2.4	Písmolijna	42
6.	Typografie a základní typografická pravidla	43
6.1	Odstavcová sazba	43
6.1.1	Odstavcová vs. Jednořádková sazba (Adobe)	44
6.2	Pravidla sazby jednotlivých znaků a prvků	44
7	Automatizované postupy při sazbě	49
7.1	Styly	49
7.1.2	Definice stylu	49
7.1.3	Druhy stylů	49
7.1.3	Vytváření a aplikace stylů	50
7.1.4	Vybrané položky z definice stylu	51
7.2	Dělení slov	53
7.3	Vzorové stránky	54
7.4	Obsahy a rejstříky	55
7.5	Vyhledávání a nahrazování	56
7.6	Makra a skriptování	58
8	Přílohy	59
A	Doporučená literatura	59
B	Stránky s informacemi ze světa DTP, pre-pressu, grafiky atd.	59
C	Vybrané glyfy a jejich ASCII kódy	60
D	Normalizované rozměry papíru	61

1 DTP, PRE-PRESS A TI JINÍ...

V posledních patnácti letech jsme jako uživatelé, čtenáři, i jako profesionálové – sazeči, grafici a typografové – svědky mohutného rozmachu počítačových technologií, které zcela transformovaly celý proces přípravy a výroby grafických výstupů, od vizitiek přes knihy až třeba po billboardy.

Ještě před několika málo lety bylo nemyslitelné či jen obtížně představitelné, že i poučený laik si může sám na vlastním počítači za pomoci cenově dostupného hardware a software připravit do tisku velkoformátové plnobarevné publikace, a to mnohdy v téměř profesionální kvalitě. To vše je dnes realitou.

Příčin tohoto vývoje je více, společným jmenovatelem je samozřejmě postupující computerizace tiskových a polygrafických technologií a lidského života západní civilizace obecně. Domácí počítač, digitální fotoaparát, skener a barevná tiskárna dnes patří k vybavení velké části domácností, nemluvě o školách všech stupňů. Ovládnutí těchto přístrojů a práce s nimi, alespoň na určité úrovni, pak často přivede uživatele k počítačové grafice, a tím časem třeba i k sazbě publikací, mnohdy přes mezičlánky typu školní časopisy, plakáty na kulturní akce nebo různou „zábavovou“ grafiku.

Tento vývoj přináší pozitivní i negativní výsledky. Určitá demokratizace oboru by mohla patřit k těm pozitivnějším. Slova jako DTP (desktop publishing, „publikování na stole“) nebo pre-press, tj. předtisková příprava dokumentů, již nepatří k výrazům z oblasti černé magie, ale k běžnému technickému výrazivu i například pracovníků v kultuře. Navíc, kolik zajímavých literárních počínů především v devadesátých letech se uskutečnilo z velké části díky tomu, že dedikovaní jedinci (mnohdy se jim říká naivní nadšenci) investovali čas a energii do přípravy různých publikací do slova a do písmene doma na stole, čímž dali termínu DTP tak trochu jiný nádech?

Obrovské zjednodušení a zlevnění tiskových technologií umožnilo a čím dál více umožňuje vydávat publikace skutečně malonákladově, a to i v barevném provedení. Vydavatelský princip Print-On-Demand, tedy tisku podle požadavku [trhu, zákazníka, prodejce], se po mnoha letech vyplněných spíše teoretickou diskuzí stává realitou. Nejnovější technologie umožňují vyrábět knihy v nákladu, který lze skutečně počítat na kusy. (Kromě technologie Computer-to-Plate pro přímou a rychlou přípravu tiskových desek při plnobarevném tisku se často tiskne také pomocí metody Computer-to-Print, tj. přímého spojení počítače s výstupním zařízením – nikoliv však stolní tiskárnou!) Ale i vydavatelství se staršími tiskovými postupy jsou samozřejmě schopná vyrábět knihy v řádu desítek kusů v ceně, která je pro koncového uživatele velmi vstřícná.

Rozšíření povšečných znalostí o fungování počítačové grafiky a počítačové sazby má však také negativní dopady. Laicizace oboru nevede vždy ke zlepšení jeho kvality, ať již v krátkodobém či dlouhodobém horizontu. Z grafiků, sazečů a typografů se tak v posledních letech stávají jacísi všezahrnující „DTP operátoři“, jak

zní tato nová pracovní pozice. Velká část těchto „DTP operátorů“ za sebou nemá klasické polygrafické nebo tiskařské vzdělání, o specializovaných vysokoškolských oborech ani nemluvě. Mnohá pravidla a metody, včetně někdy těch základních, se pak učí takřkajíc za pochodu. Nejedni z nich navíc, byť samozřejmě nechtěně, nahradili pracovníky, kteří navzdory mnohaletým zkušenostem nebyli schopni nebo ochotni se přeorientovat na nové digitální technologie, avšak právě jejich zkušenosti a váha názoru často chybějí.

Pomalé pronikání kdysi specializovaných oborů mezi (ne)poučené laiky vede navíc často k tomu, že k přípravě tiskových podkladů jsou používány nevhodné programy, zapomíná se na desetiletými ověřená typografická pravidla, používají se univerzální „windowsovská“ písma a upřednostňuje se ekonomická stránka věci před estetickou. Tento trend je bohužel viditelný všude, od regionálního tisku přes mnohé odborné publikace až po beletrii, která má čtenáře doprovázet celý život.

Jednou z cest, jak se vypořádat s tímto trendem, je i snaha některých škol zařadit do svého curricula předměty, které by studenty různých oborů o dané problematice alespoň částečně poučily. A to i tehdy, kdy jde o obory, které se světem DTP a pre-pressu primárně nesouvisejí. Pro takový kurz je připraveno i toto skriptum, jehož skladba a obsah odpovídají požadavkům kladeným na studenty oboru *Česká filologie se zaměřením na editorskou práci ve sdělovacích prostředcích*. Snaha o prakticky zaměřené předměty v rámci studia odpovídá také faktu, že znalosti DTP a typografie jsou vítaným plusem pro budoucí uplatnění absolventů vysoké školy humanitního směru.

Je samozřejmě pravda, že na knižním trhu se v posledních patnácti letech objevila řada odborných publikací, které se věnují jak celkovému, tak dílčímu pohledu na tuto problematiku nebo třeba jen jednotlivým programům. Vždyť málokterá část knižního trhu zažila po roce 1990 takový boom jako počítačová a technická literatura, která s tématy zde probíranými souvisí. Faktem na druhé straně zůstává, že většina těchto publikací je finančně nákladná a v knihovnách, ať již univerzitních nebo jiných, málokdy v pravou chvíli dostupná. To bylo dalším impulzem vedoucím k sepsání tohoto textu. Věříme, že bude užitečnou pomůckou všem zájemcům.

Upozorňujeme ještě na fakt, že s přihlédnutím k cílové skupině i záměru autora byly mnohé prvky zde probírané problematiky, především technického a technologického rázu, dosti zjednodušeny. Prosíme odborníky a specialisty o shovívavost.

2 SOFTWAREVÉ VYBAVENÍ

Softwarové vybavení, se kterým se student, uživatel či pracovník DTP zaměření může setkat, lze obecně rozdělit do čtyř kategorií: textové editory, textové procesory, samotné programy pro sazbu (DTP programy, „lámací programy“) a skupina programů pro úpravu grafiky.

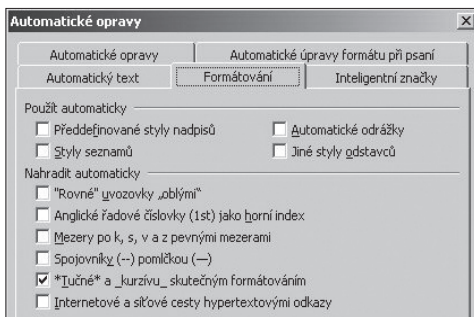
Kromě toho lze v každém grafickém studiu i v jednotlivých počítačích grafiků nalézt řadu pomocných programů, např. na prohlížení a správu fontů (počítačových písem), správu a katalogizaci fotografií a jiných obrázků či jejich konverzi a hromadné zpracovávání, software pro práci s hardwarovým vybavením (skenery, tiskárny, plottery atd.) a mnohé další. K některým z nich se dostaneme dále v příslušných kapitolách.

2.1 Textové editory

Textové editory jsou nejjednoduššími programy pro zpracování textu. Jejich typickým a na světě nejspíše nejrozšířenějším příkladem je tzv. Poznámkový blok (Notepad) ve Windows. Již z toho je zřejmé, že textové editory mají pouze velmi základní nebo i vůbec žádné nástroje pro grafickou úpravu textu. Nejjednodušší textové editory pracují v režimu tzv. plain textu (plain = jednoduchý).

Bylo by však chybou se domnívat, že editory jako takové nemohou vůbec sloužit k formátování textu. Existují ustálené normy a postupy popisující, jak lze text formátovat za použití určitých uzůálních formátovacích značek. Takto meta popsany či označovaný plain text při výstupu (tisk, zpracování na obrazovce) obsahuje grafické prvky a formátování, které většina uživatelů jinak dosahuje pomocí textových procesorů nebo i DTP programů. Značkovací systémy mohou být okazionální, jednorázové, nebo uzůální, normované – z těch je nejznámější hypertextový popis textu (HTML a jeho vyšší vývojové stupně) a v oblasti DTP systém založený na formátovacím jazyce TeX.

Okazionální formátovací značky používané pro plain text využívá dnes již málokdo a pokud ano, pak nejspíše v prostředí, kde si toho ani není vědom – na chatovacích systémech s omezenou možností využití grafiky atd. Správná interpretace těchto značek závisí na uživateli. Výraz **slovo** pak musí příjemce sám dekodovat jako **slovo**, resp. formátování `_slovo_` chápat jako *slovo*. Tyto kdysi všeobecně užívané značky přežívají mj. i jako relikty ve volbě Automatický formát (menu Formát, položka Automatický formát) Microsoft Wordu, kde je možné libovolný text



2.1 – Automatické formátování

(předpokládejme, že napsaný v plain textu) převést na jeho „moderní“ grafickou podobu. Viz obr. 2.1.

Normativní systémy jako HTML používají mnohem sofistikovanější systémy popisu grafické podoby a struktury stránky. (Ponechejme v tomto případě zcela stranou, že HTML a jiné jazyky slouží k mnoha dalším věcem, zaměříme se pouze na grafickou podobu a její problematiku). Jde o metajazyk, který pomocí ustálených značek (tagů) popisuje výslednou podobu stránky. Slovem „výslednou“ zde rozumíme její zobrazení na monitoru či případně jiných výstupních zařízeních poté, co jiný program, který slouží jako překladáč (interpret) daného metajazyka, zpracuje HTML kód a uživateli nabídne grafický výsledek.

Jinými slovy kód:

```
<p>Filmovost nicméně Enquistův <b><i>Hamsun</i></b> nezapře především svou vizuálností. <i>Říšský komisař Terboven</i> pak zamíří do krytu, betonového bunkru dvakrát dva metry, který mu vždycky připadal malý.</i>
```

příslušný překladáč (Firefox, MS Internet Explorer, Opera a řada jiných včetně mnoha textových procesorů) zobrazí jako:

Filmovost nicméně Enquistův ***Hamsun*** nezapře především svou vizuálností. *Říšský komisař Terboven* pak zamíří do krytu, betonového bunkru dvakrát dva metry, který mu vždycky připadal malý.

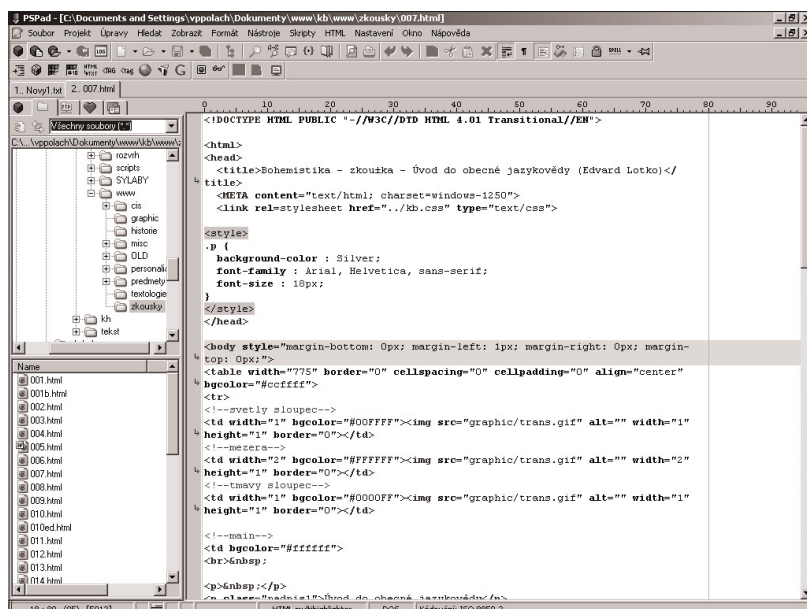
Je tedy zřejmé, že mnohé z užívaných tagů lze dekódovat poměrně jednoduše při využití znalostí anglického jazyka: značka <p> znamená odstavec, paragraph; značka <i> kurzívu – italicu, italic, tučné písmo, bold atd. HTML, XML a jiné jazyky spolu s pomocnými nástroji lze ovšem využít k opravdu komplexním a komplikovaným grafickým úpravám, při jejichž interpretaci si již laik nemusí a pravděpodobně nemůže vystačit bez hlubší znalosti kódu (kódů).

Jako poslední normativní systém zmiňme plně vyvinutý typografický systém TeX (čti [tech]) a jeho různé varianty, z nichž mezi nejrozšířenější u nás patří patrně LaTeX (ačkoliv „výslovnost“ [latex] či [latech] je stále populární, správná je samozřejmě [lej-tech]). Jde o komplexní systém, rozšířený převážně na Linuxu a využívaný jak k jednoduché knižní, tak i ke komplikované sazbě např. matematických rovnic a vůbec technické literatury. Pro tento účel byl koneckonců vyvinut a dodnes jde o systém prosazovaný na mnoha technických univerzitách a fakultách včetně českých. TeX je, podobně jako HTML, metajazyk, který popisuje výslednou grafickou podobu zamýšleného dokumentu. Primárně tedy nepracuje v prostředí WYSIWYG (viz str. 10), byť má již dnes mnoho pomocných programů, které jsou orientovány na zjednodušení práce v tomto grafickém prostředí. Ortodoxní uživatelé TeXu, lze-li takto paušalizovat, však stále preferují práci v prostředí plain textu, jak jej známe z textových editorů.

Dodejme ještě, že v nejjednodušších editorech dnes pracuje již málokdo. Plain text může někdy sloužit jako dobrý základ pro další grafické zpracování v proce-

sorech nebo v sázecích programech, mnohé editory mají však natolik omezené palety dalších nástrojů, že práce s nimi nebývá zrovna „uživatelsky příjemná“.

I to je jedním z důvodů, proč se během posledních let objevila řada editorů „vyšší řady“ poskytujících příjemné grafické prostředí a mnohé nástroje k usnadnění práce. Většina z nich byla vyvinuta primárně pro tvůrce webových stránek v HTML, XML a jiných jazycích a umí například zvyrazňovat jednotlivé značky (tagy) v textu a odlišit je tak od vlastního obsahu stránek. Jednou z dalších výhod těchto programů je také jejich cena – jsou buď zcela volně šiřitelné (freeware, GNU licence, Open Source) nebo zdarma dostupné alespoň v různě omezených verzích. Jako příklady lze uvést např. PS Pad či Notepad++.



2.II – Pokročilý textový editor PS Pad (a HTML kód)

Otázkou zůstává, kam řadit programy typu WordPadu, tedy jakousi zjednodušenou verzi Wordu, kterou Microsoft distribuuje spolu se systémem Windows. Jde o program s grafickým rozhraním a prvky WYSIWYG zobrazení, ovšem s přesto omezenými formátovacími možnostmi. Nepříliš šťastně se musíme spokojit s tím, že jde o tranzitní krok směrem k textovým procesorům, nedosahující však na některé základní prvky, z nichž mezi nejdůležitější lze počítat fakt nemožnosti zobrazení celé stránky a práce s ní. (Funkci tzv. Náhledu (menu Soubor, položka Náhled) záměrně opomímám, neboť jde vlastně o nadstavbu, navíc nepraktickou.)

Dalším programem, který je nutné řadit spíše k editorům, byť nabízí základní a prajednoduché prvky grafického zobrazení, je legendární, ovšem dnes již zcela nevyhovující český editor Text602.

2.2 Textové procesory

Textové procesory jsou programy, ve kterých lze, alespoň teoreticky, plně zpracovávat (viz anglické *to process* = zpracovat) grafickou podobu výsledného textu. Textové procesory by měly splňovat podmínky WYSIWYG principu.

WYSIWYG je akronym z anglického *What You See Is What You Get*, česky poněkud kostrbatěji tedy „Co vidíte [rozuměj na obrazovce], to dostanete [při výstupu, tedy většinou při tisku].“ Ačkoliv mnozí kritikové a mnozí profesionální grafici mají k tomuto termínu ironický odstup a považují jej spíše za marketingové heslo, částečně oprávněně, nelze opomenout, že to byl mimo jiné právě přechod od uživatelsky méně příjemných textových editorů k WYSIWYG procesorům, co umožnilo rozšíření PC od domácností po kanceláře a jejich využití ke zpracování textu. To, že si dnes mnozí pletou textové procesory se sázecími programy, a vznikají tak knihy připravené ve Wordu a jeho klonech, často v otřesné typografické kvalitě, je ovšem hlavně chybou a omylem uživatelů.

Co chybí textovým procesorům pro faktickou využitelnost k typografii, je mj. poměrně laxní přístup ke zpracování, zobrazení a manipulaci s grafikou; absence profesionálních funkcí pro nastavení a správu barev u barevných dokumentů; velmi volná práce s písmy (fonty), jako je vytváření tzv. falešné kurzívy (viz kapitola o písmu); nemožnost pracovat se samostatnými objekty v rámci strany a mnohé jiné.

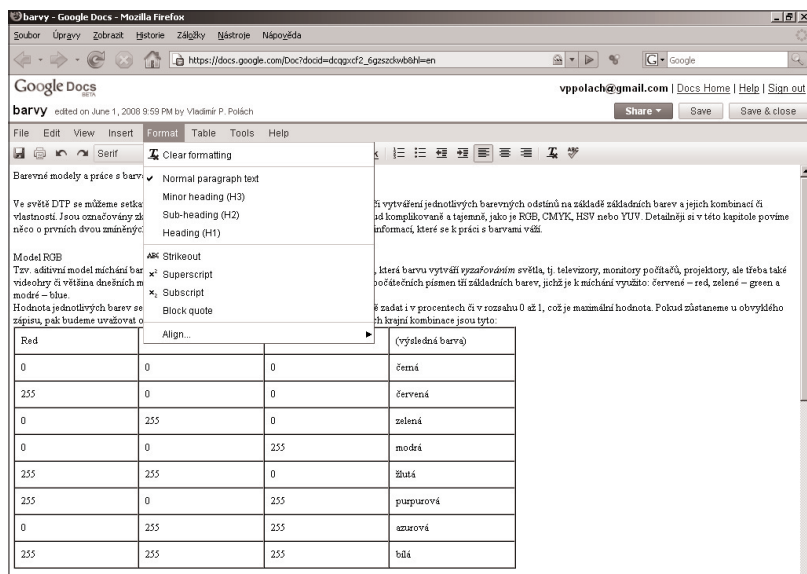
Na poli textových editorů dnes bez diskuze vládnou různé verze Microsoft Wordu, prodávaného jako součást balíku Office. Microsoft také nabízí jakýsi „menší“ kancelářský balík s názvem Microsoft Works. Poslední verze MS Office (2007, někdy omylem nazývaná Vista; ve verzi pro Macintosh je ovšem označována dle data vydání jako 2008) nicméně vyvolala jistou kontroverzi svým novým DOCX (přesněji Office OpenXML) formátem dokumentů. Ten nabízí jen omezenou kompatibilitu směrem k jiným textovým procesorům, včetně vlastních starších produktů (pro starší verze Office byl Microsoft nakonec nucen připravit a vydat převáděcí filtry, které umožní otevřít formát DOCX i ve starších balících Office, ne však starších než verze 2000).

Zdarma šiřitelnou a stále populárnější alternativou (zatím s nejedním minusem či chybějícími funkcemi, ale například s výborně fungujícím exportem do PDF) je pak OfficeWriter balíku OpenOffice. Tuzemská varianta balíku kancelářských aplikací, 602Suite (od výrobce Software 602, který kdysi ovládl české počítače se svou T602), se již přestala samostatně vyvíjet. Je nicméně možné si za symbolický poplatek stáhnout „OpenOffice verze Software602“, která tuzemským zájemcům nabízí některé funkce navíc. Podobně populární StarOffice vychází z OpenOffice,

do které firma Sun Microsystem dodává nadstavbové funkce, jde ale o placený produkt.

Ve světě, převážně v USA, býval rozšířený procesor WordPerfect, dnes patřící koncernu Corel, o kterém se ještě budeme bavit; pamětníci vzpomenu i na první rozšířený textový procesor AmiPro (později WordPro) či mnohé jiné varianty.

Zatím v začátcích je projekt webových (web-based) textových editorů a vůbec celých kancelářských balíků a jiných programů, z nichž největší budoucnost lze předpokládat u projektu Google Docs (Google Dokumenty v české verzi) – pro ty, kdo mají účet u Google, dostupný na adrese <http://docs.google.com> (obr. 2.III).



2.III – Google Docs

2.3 DTP programy

Na poli profesionálních, plnohodnotných DTP programů vládne již několik let dvojice konkurentů – InDesign od nejznámějšího výrobce programů spojených s grafikou a typografií, firmy Adobe, a QuarkXPress firmy Quark.

QuarkXPress má za sebou dlouhou a úspěšnou historii. Vyvíjen od konce 80. let, v letech devadesátých se stal rozhodně nejrozšířenějším profesionálním DTP programem, a to jak na Apple Macintosh platformách, tak od verze 3.1 také pod Windows. Zatím poslední verze (8) je certifikována jak pro poslední Windows Vista (a nepoběží na nižších než Windows XP), tak pro nejnovější Mac OS X.

Jeho v současnosti největší konkurent InDesign byl uveden na trh až v roce 1999, mohl však navazovat na zkušenosti z vývoje a využití dnes již ukončeného Aldus/Adobe PageMakeru. Momentálně existuje ve verzi 5, známější ovšem jako verze CS3

poté, co se InDesign začal zahrnovat spolu s jinými grafickými programy stejného výrobce do balíku nazvaného Adobe CreativeSuite, který existuje ve své třetí vývojové verzi. Pracovní prostředí programu si lze prohlédnout na obrázku 2.IV.

Vzhledem k tomu, že projekt *Inovace studijních programů bohemistiky s ohledem na potřeby soudobé informační společnosti*, v rámci kterého je vytvořen i tento text, nám umožnil zastřešit výuku typografie a DTP pořízením právě InDesignu CS3, budou se příklady a ukázky z praktické oblasti v tomto skriptu vztahovat k tomuto programu. Většinu těchto funkcí by však uživatel samozřejmě našel i v jiných, konkurenčních programech včetně QuarkXPressu.

Oba tyto programy poskytují uživateli de facto vše, co je potřeba k vytváření tisků a tiskovin všech kategorií (za předpokladu, že podklady pro sazbu – obrázky, písma atd. – jsou již v pořádku a ve finální podobě). Jejich nevýhodou pro soukromé či studijní využití je bohužel cena.

Jediným zástupcem volně šiřitelného programu z této kategorie je Scribus, průběžně vyvíjený a doplňovaný o nové funkce a možnosti, a dostupný navíc pro Windows, Mac OS i Linux.

Mezi zástupce z této kategorie také patří některé programy, jejichž vývoj byl již ukončen nebo jejichž rozšíření je spíše marginální. Do první skupiny patří i u nás poměrně populární Adobe PageMaker, který se zastavil na verzi 7 (ta ale již nebyla počestněná). Mezi programy méně rozšířené a často určené pro specifické dokumenty (např. rozsáhlou technickou dokumentaci) pak patří třeba Corel Ventura nebo Adobe FrameMaker. Corel Ventura je přímý pokračovatel průkopnického programu Ventura Publisher, který umožnil pracovat s grafickým prostředím i na počítačích, které podporovaly pouze „negrafický“ systém MS-DOS na počátku devadesátých let dvacátého století.

2.4 Ostatní software

Je samozřejmé, že do vybavení DTP pracoviště patří také programové vybavení pro práci s grafikou, písmy (fonty), specializovaným hardware atd. Vzhledem k tomu, že tyto programy se vážou také k jednotlivým tématům této příručky, je jejich přehled a případná stručná charakteristika zařazena dále s výjimkou OCR programů.

Zkratka OCR znamená *optical character recognition* [software]. Jde o takové nástroje, které umí pomocí skeneru již jednou vytištěný text převést zpátky do jeho digitální podoby – a to ne jako bitmapovou grafiku (tj. „digitální obrázky“), ale přímo do formátu počítačového písma. S OCR software se lze setkat poměrně často, v DTP se tyto programy nejčastěji využívají při vytváření podkladů pro reedice již jednou publikovaných tisků apod. U nás byl dlouho na špici oblíbenosti a využitelnosti program Recognita (v posledních verzích jako Recognita Omnipage), dnes má častého konkurenta v Abbey FineReaderu. OCR software v různém stupni dokonalosti a možností nastavení se často přikládá ke stolním skenerům, a to i k těm v nízké cenové relaci, určeným pro kancelářské nebo základní grafické využití.

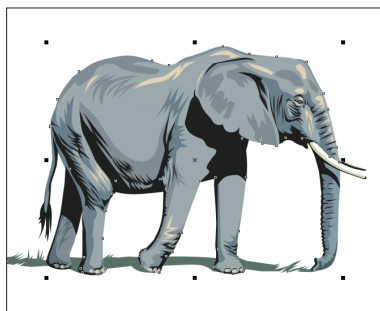
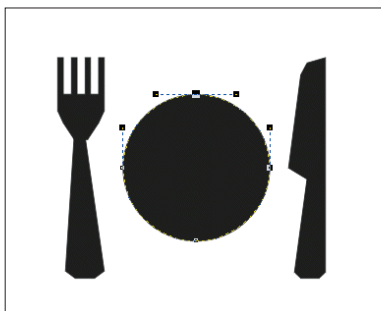


3 GRAFIKA

Grafické prvky, se kterými se můžeme setkat při sazbě dokumentů, lze obecně rozdělit na grafiku vektorovou a bitmapovou (také „rastrovou“). V této kapitole si ukážeme základní rozdíly mezi nimi, přiblížíme si nejpoužívanější programy pro práci s grafikou, vybrané formáty grafických souborů a přiblížíme si i některé technické detaily.

3.1 Vektorová grafika

Ačkoliv lehce abstraktní definice vektorové grafiky zní, že jde o grafiku definovanou na základě geometrických tvarů, jejich vlastností a vztahů mezi nimi, každý z nás si ji nejspíše lépe vybaví, řekneme-li, že vektory se používají při tvorbě piktoqramů, log, symbolů a podobných obrazců. Asi nejrozšířenějším příkladem využití vektorové grafiky na počítačích je však samotné počítačové písmo (existenci bitmapových fontů v tuto chvíli vědomě ignorujeme). Také např. grafy, mapy, tabulky a vůbec infografika, známá z moderních časopisů a jiných tiskovin, je vytvářena pomocí vektorové grafiky.



3.1 Typická ukázka nejjednodušší a složitější vektorové grafiky. Povšimněme si bodů a ovládacích křivek, pomocí kterých jsou jednotlivé prvky grafiky definovány.

Nenechme se těmito příklady ovšem zmást, vektory mohou vytvářet také komplexní obrazce jak co do složitosti, tak práce s barvami (přechody, textury) a mají opravdu komplexní využití. Vektorová grafika navíc může simulovat i prostorové vyjádření – i laik nejspíše někdy viděl tzv. 3D modelování (techniky, architektury či třeba počítačových filmových triků; tzv. CAD systémy). Vektorová grafika je také vhodnější pro tvorbu animací a jeden z vektorových grafických programů, Adobe Flash, byl vytvořen primárně pro vytváření vektorových animací, především pro internet.

3.1.1 Výhody vektorové grafiky

Oba typy grafik nelze samozřejmě jednoduše srovnávat – oba jsou primárně určeny pro jiné využití. Vektorová grafika díky tomu, že je definována matematicky (geometrické tvary a vztahy mezi nimi), ve výsledku:

- vytváří co do velikosti v bajtech („byte“ je jednotka množství dat, rozlišujeme od bitů, což je jednotka informace; s bity se v grafice setkáváme třeba u určování hloubky barev, jak ještě uvidíme) malé soubory na disku
- díky matematické „velikosti“ lze každou vektorovou grafiku téměř libovolně zmenšovat a zvětšovat, aniž by docházelo k deformacím (jinými slovy, vektorová grafika není definována v definitivním rozměru)
- jednotlivé prvky (objekty) grafiky lze dále upravovat a redefinovat – jsou tvořeny sítí bodů, křivek, výplně atd., která je součástí flexibilní definice obrázku; objekty je také možné otáčet, přesouvat atd.
- je vhodná pro 2D a 3D modelování, včetně generování textur a stínů

3.1.2 Softwarové vybavení

O pozici nekorunovaného krále při práci s vektorovou grafikou již po více než dekádu soupeří CorelDRAW a Adobe Illustrator.

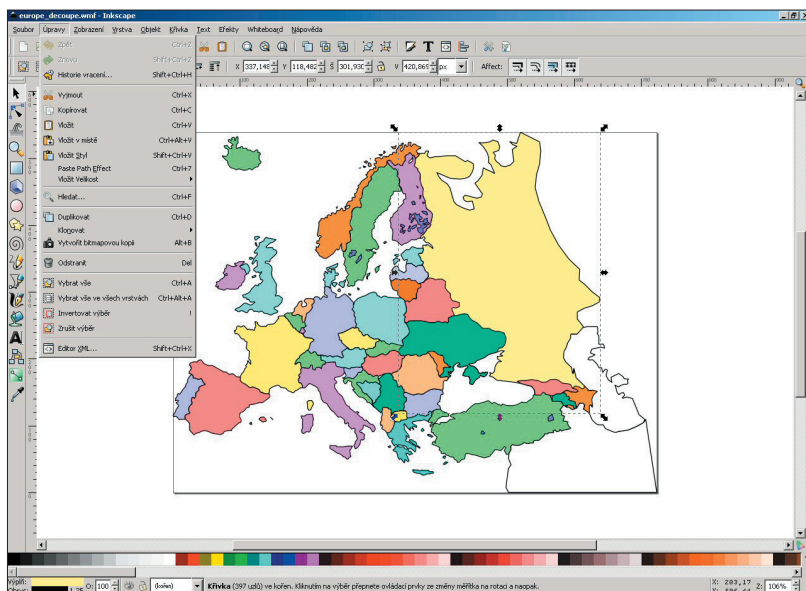
CorelDRAW, vyvíjený od začátku 90. let, se v době psaní této publikace vyšplhal na verzi s pořadovým číslem 14, je ovšem nabízen pod novým marketingovým označením X4. Stejně jako mnohé jiné programy je i CorelDRAW nabízen jako součást většího balíku programů – CorelDRAW Graphic Suite (neplést s Corel WordPerfect Office, což je balík kancelářských programů podobně jako MS Office). Za zmínku stojí i fakt, že nejnovější CorelDRAW funguje již jen na platformě Windows XP a vyšší.

Adobe Illustrator, s historií sahající dokonce až do 80. let a vyvíjený nejdříve stejně jako jiné produkty firmy Adobe pro platformu Apple Macintosh, se momentálně prodává ve verzi 13, resp. CS3 (jako součást Creative Suite). Na rozdíl od své největší konkurence je Illustrator stále podporován pro Mac OS X, operační systém pro Apple Macintosh, ve Windows vyžaduje také minimálně platformu Windows XP.

Z ostatních komerčních programů pro práci s vektorovou grafikou lze zmínit alespoň Adobe (dříve Macromedia) Freehand, Microsoft Visio (především pro tvorbu diagramů, součástí některých vydání MS Office) či Xara Xtreme (jednu dobu jako CorelXARA), jejíž linuxová verze je distribuována zdarma.

Většina výše uvedených programů má pro soukromé či studijní využití jednu zásadní nevýhodu, a tou je pořizovací cena (ačkoliv je nutné uvést, že tzv. edukační licence pro vysoké školy a podobné subjekty jsou dostupné za zlomek plné ceny). Dostupné jsou však i volně šiřitelné programy, jako je OpenOffice Draw – tedy součást balíku OpenOffice – nebo pravděpodobně nejrozšířenější volně šiřitelný vektorový editor Inkscape (viz obr. 3.11), průběžně inovovaný a vyvíjený. Jeho nevýhodou ovšem je ukládání grafiky výhradně ve formátu SVG, výhodou naopak podpora všech tří základních systémových platforem – Windows, Linuxu a Mac OS.

Nejzákladnější funkce pro práci s vektorovou grafikou v sobě obsahují i mnohé DTP programy.



3.II Inkscape

3.1.3 Nejrozšířenější formáty

AI	Nativní formát Adobe Illustratoru, slouží také jako určitý standard pro výměnu grafiky, především u profesionálních grafiků.
CDR	Nativní formát CorelDRAW.
SVG	Scalable Vector Graphics je formát vyvinutý konsorciem W3C (vývoj webového rozhraní internetu) pro univerzální použití, primárně na internetu. SVG může být animované.
WMF	Windows Meta File. Formát primárně vektorové (ovšem s možnostmi zahrnout bitmapu) grafiky pro Microsoft Windows. Lze užít poměrně univerzálně na platformě Windows, ovšem s technickými omezeními danými dobou jeho vzniku (začátek 90. let); jde vlastně o metaformát, lze jej využít pro přenos různých vektorových dat přes schránku Windows.
EMF	Enhanced Metafile. Technicky pokročilejší forma (32bitová grafika) původních WMF.
PS	PostScript. Viz níže.
EPS	Encapsulated PostScript. Viz níže.
PDF	Portable Document Format. Viz níže.

3.1.4 PostScript a PDF

PostScript je definovatelný jako jazyk popisu (grafiky) stránky, odpovídá tedy definici vektorové grafiky, nicméně dle potřeby zahrnuje i případné bitmapové prvky; jde o metaformát. Postsriptový soubor tedy „popisuje“ výsledný vzhled jakéhokoliv grafického prvku poté, co bude zpracován výstupním zařízením, ať již je to monitor či tiskárna. Vyvíjený již od osmdesátých let, v letech devadesátých se stal de facto profesionálním standardem pro grafická studia a tiskárny. Aniž bychom zacházeli do technických detailů, stačí shrnout, že grafická předloha, uložená (vygenerovaná) jako postsriptový soubor, mohla sloužit k profesionálnímu tisku na všech zařízeních, která ovládala tento jazyk. (Podíváme-li se do katalogů prodejců stolních tiskáren z 90. let, uvidíme ještě dělení na postsriptové a nepostscriptové tiskárny; ty druhé byly určeny pro domácí použití.)

Postsript sám, vzhledem k mnoha technickým (softwarovým i hardwarovým) potížím, které jeho využití s sebou přinášelo, je v posledních letech vytlačován novým univerzálním přenosným formátem – PDF. Přesto je stále možné se setkat se soubory ve formátu PS (*.ps), především u takových grafických prvků, kde zhotovitel, zadavatel či grafik chce zaručit kvalitu a neměnnost výstupu – loga, logotypy, ustálené piktogramy atd.

Některé starší grafické a DTP programy nedokázaly zobrazit postsriptové soubory na obrazovce a místo výsledné grafiky ukázaly jen šedý rámeček s názvem souboru. Výsledek byl zřetelný až při tisku na postsriptovou tiskárnu (nebo např. poté, co bylo z postsriptových podkladů vygenerováno PDF).

(Přesněji: existují tzv. překladače, interpretery postsriptu, známé jako RIPPY – Raster Image Processor(s); odtud slovní výraz „ripovat“, „ripovat postsript“, který patří do hanáky počítačových grafiků.)

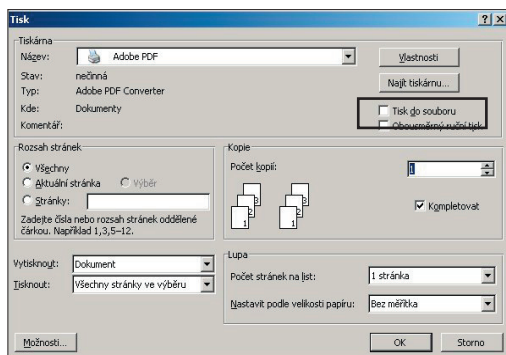
PostScript jako jazyk byl vyvinut do aktuální verze 3 (PostScript 3, předchozí stupně nesly ještě označení „Level“).

EPS, Encapsulated PostScript, je variantou postsriptových souborů, obsahujících dostatečné informace na to, aby jednotlivé programy mohly analyzovat jejich obsah (vygenerovat grafický popis stránky), aniž by přitom musely zpětně dekodovat („ripovat“) celý postsriptový obsah souboru, což je poměrně náročné na výkon počítače. Kromě toho, což je z praktického hlediska výhodné, EPS soubory v sobě obsahují jednoduchý náhled vlastního obsahu, většinou ve formě bitmapového obrázku. (Náhledy bylo teoreticky možné připojovat již k původním postsriptovým souborům.)

Protože samotný PostScript je pro počítač, respektive uživatele, nečitelný (jde o „slovní“, kódový popis grafiky), existují programy, které z PostScriptu vytvářejí PDF soubory. Zatímco skupina produktů označována a distribuována jako Adobe Acrobat (obsahuje mj. Adobe Reader, Adobe Professional či původní a dodnes mnohými používaný Adobe Distiller) patří mezi komerční, placené produkty, GhostScript nebo PDF Creator jsou volně šiřitelné.

PDF, Portable Document Format, je formát vyvinutý firmou Adobe a používaný dnes jako univerzální formát výměny grafických informací. Na rozdíl od postskriptu nejde o programovací jazyk, ale „pouze“ o formát dokumentů (podobně jako již zmiňovaný formát WMF atd.), jednoduše zobrazitelný a zpracovatelný. Jde primárně opět o vektorový formát, umožňující ovšem zahrnout i bitmapové obrázky v kvalitě, kterou lze uživatelsky nastavit a ovlivnit. PDF dokumenty jsou navíc schopné komprese bitmapových dat. PDF soubory jsou přenositelné („portable“) mezi jednotlivými počítači i mezi různými platformami.

„Původní“ způsob vytváření PDF dokumentů přes tisk do postskriptového souboru a následně extrahování (k tomu sloužil Adobe Distiller – dodnes se tak v češtině grafiků „distilují pédeéčka“) PDF z něj lze dnes nahradit jednoduššími způsoby. Ne-nápadná volba „tisk do souboru“, kterou dodnes nabízejí de facto všechny programy pracující pod Windows (viz obr. 3.III), je však kromě jiného reminiscencí na tento postup. (Zvídavým uživatelům tímto také naznačujeme, jakým způsobem vytvořit PDF z libovolného programu pro Windows.)



3.III Funkce Tisk do souboru

Dnes, požadujeme-li výstup do PDF, můžeme volit programy, které mají implementovaný přímý export do tohoto formátu, jako jsou programy ze skupiny OpenOffice nebo InDesign, případně využít pro export některé volně šířené či placené specializované programy zmíněné výše.

3.1.5 Trasování obrázků

Trasováním (z angl. *to trace* – obtažnout, obkreslit) obrázků myslíme jejich vektorizaci, tj. převádění bitmapových předloh do vektorového formátu. Tento poměrně atypický postup lze někdy s úspěchem využít. K trasování můžeme využít specializované programy, jako je např. Corel TRACE (dnes ve verzi Corel PowerTRACE X4), nebo příslušné vestavěné funkce některých jiných programů, třeba Adobe Illustratoru.

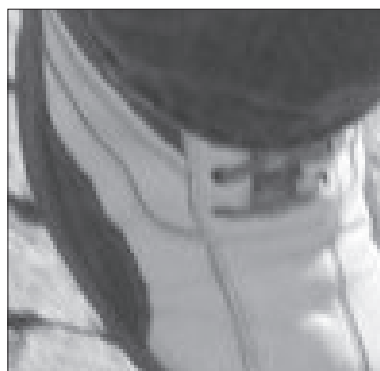
3.2 Bitmapová grafika

Bitmapovou grafiku lze jednoduše popsat jako grafiku složenou z matrice (ras-tru) jednotlivých bodů (pixelů), z nichž každý nese informaci o své barvě, jas a kontrastu, a jako celek dávají výsledný obraz. Na rozdíl od vektorové grafiky je výsledný obrázek definován dvourozměrnou maticí, jejíž počet bodů je konečný,

což limituje možné další operace s ním. Opět snad nejjednodušší – typickým příkladem bitmapové grafiky je digitální fotografie. Na pomyslném druhém pólu komplexních barevných fotografií stojí jednoduché černobílé kresby, kterým se pro podobnost s kreslicí technikou říká pérovky (mnohdy jde o neskenované obrázky, které opravdu vznikaly kresbou perem nebo podobnou technikou, viz obrázek 3.IV až 3.VI). V tiskařském profesním slangu se lze také setkat s pojmem „autky“. Auto-pérovky jsou vlastně bitmapové obrázky ve stupních šedi, simulující kresbu uhlem či tužkou (ale lze sem zařadit i fotografie a vůbec obecně tónové černobílé obrázky).



3.IV Bitmapa v odstínech šedi



3.V Zvětšený detail z obrázku 3.IV ukazuje matici jednotlivých bodů



3.VI Pérovka

3.2.1 Výhody (a nevýhody) bitmapové grafiky

- bitmapový obrázek lze snad přirovnat k malířovu plátnu: má definovanou velikost, ale jinak skýtá obrovské možnosti co do obsahu; jen díky bitmapám lze dosáhnout opravdové fotorealističnosti

- vzhledem k objemu dat, které obsahují jednotlivé obrázky (obrázek o velikosti 640×480 pixelů má celkem 307 200 bodů, které mohou obsahovat i mnohabitové informace o barvě – viz barevná hloubka níže), vytváří bitmapová grafika relativně velké soubory v počítači (obyčejné fotografie bez komprese v řádu megabajtů)
- bitmapová grafika, definovaná maticí bodů, je při svém zobrazení závislá na tzv. rozlišení výstupního zařízení; čím vyšší je výstupní kvalita, tím relativně menší dostaneme výsledek (viz níže)
- nevýhodou je omezená nebo nulová možnost uchovávat bitmapový obrázek jako soubor jednotlivých a samostatných částí, které by možné dále upravovat; např. jednou vložené písmo se stává maticí bodů, nikoliv písmeny, které by bylo možné dále přepisovat; tuto nevýhodu částečně řeší některé specializované formáty.

3.2.2 Softwarové vybavení

Jednoznačně nejrozšířenějším profesionálním programem a svého druhu standardem pro práci s bitmapovou grafikou je Adobe Photoshop. Vyvíjený od konce osmdesátých let a vržený na trh v roce 1990, jeho poslední oficiální verze nese číslo 10 (resp. CS3 při integraci do balíku CreativeSuite; existuje také verze Extended). Photoshop, dostupný samozřejmě i ve verzi pro Apple Macintosh a s oznámenou podporou pod Linuxem, je také velmi dobře propojen s jinými produkty Adobe, z nichž specifické místo při zpracování bitmapové grafiky zaujímá Adobe ImageReady. Jde o program vyvinutý primárně pro zpracovávání webové grafiky, de facto do jisté míry odlehčenou variantu samotného Photoshopu. V poslední verzi CreativeSuite již ImageReady chybí a je nahrazen programem Fireworks, který Adobe odkoupilo od konkurenční firmy Macromedia.

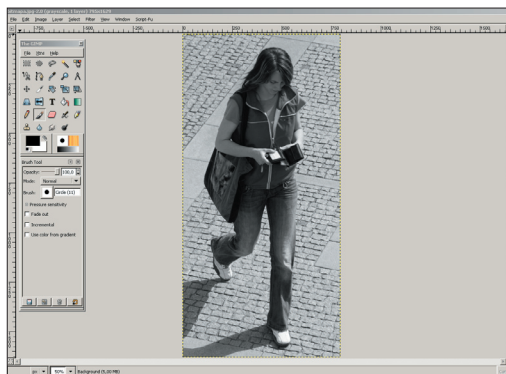
Hlavní konkurent firmy Adobe, Corel, má samozřejmě své koně ve hře. Paralelním programem k vektorovému CorelDRAW byla vždy Corel Photo-Paint, dnes dostupná jako součást balíku CorelDRAW GraphicSuite (ve verzi X4, resp. 14; pro Macintosh 11). Corel Photo-Paint je ale fakticky bližší Adobe ImageReady, resp. Fireworks, tedy relativně jednodušším programům, zaměřeným spíše na práci s webovou grafikou, případně vytváření ilustrací atd.

Corel svou programovou nabídku v roce 2004 rozšířil také o program Corel Paint Shop Pro (původně od firmy Jasc Software), dnes dostupný ve verzi označené X2, ale i samostatně, tj. mimo corelovskou GraphicSuite. Paint Shop Pro má implementovanou také podporu vektorové grafiky a může pomocí rozšiřovacích modulů emulovat konkurenční Photoshop, je však dostupný pouze ve verzi pro Windows.

Protože pole bitmapové grafiky, především pak digitální fotografie, je otevřené téměř všem uživatelům počítačů a internetu, existuje také mnoho programů buď zcela volně šiřitelných, nebo v různých „odlehčených“ verzích. Ty jsou také často součástí softwarového vybavení některých počítačových zařízení spjatých se zpracováním grafiky, nejčastěji skenerů atd. Mezi populární tituly patří např.: IrfanView, Picasa

nebo zcela zdarma dostupný Gimp, vyvíjený pod GNU licenci včetně jakéhosi vizuálního klonu Adobe Photoshopu s názvem Gimpshop.

Díky popularitě Microsoft Office je také poměrně rozšířený Microsoft Photo Editor, od verze 2007 pak nahrazený Microsoft Office Picture Managerem.



3.VII Gimp(Shop)

3.2.3 Nejrozšířenější formáty

PSD	Nativní formát Photoshopu (Photoshop Document), se kterým ovšem umí pracovat většina bitmapových editorů. Výhodou PSD formátu je mj. možnost uložit obrázek nikoliv jako jednodílnou matici bodů, ale jako souhrn objektů, se kterými lze dále pracovat. Variantou PSD pro soubory nad 2 GB je formát PSB.
PSP	Nativní formát (Corel) Paint Shop Pro.
PCD	Kodak PhotoCD – speciální formát digitálních fotografií vyvinutý firmou Kodak a používaný pro distribuci fotografií na speciálním médiu CD-ROM (tzv. Photo CD). Jeden PCD soubor většinou obsahuje až pět různých rozlišení jednoho obrázku včetně takových, která jsou použitelná pro profesionální tisk.
BMP	Jeden z nejstarších formátů pro ukládání bitmapové grafiky, známý také jako DIB formát. BMP soubory jsou ukládány bez komprese. Často slouží jako grafické prvky pro Windows (již od verze 3.0) či nejstarší Mac OS/2.
GIF	Graphics Interchange Format. Pravděpodobně nejrozšířenější formát specifických grafických prvků (ikony, reklamní bannery) na internetu díky schopnosti bezztrátové komprese a především animace. Jeho nevýhodou je omezení na maximální paletu 256 barev, kvůli kterému není schopen opravdové fotorealismu.
PNG	Portable Network Graphic. Nástupce formátu GIF pro univerzální použití na internetu, odstraňující barevnou limitaci svého předchůdce. Samotný formát PNG patří mezi nejlépe vytvořené standardy dnešního elektronického světa a má širokou škálu potenciálního využití včetně pre-pressu a světa DTP.

RAW	Jinak neupravená data importovaná přímo ze snímacího čipu u digitálních fotoaparátů či kamer. Data v RAW formátu, nazývaná někdy „digitálním negativem“, je nutné dále zpracovat v příslušném bitmapovém editoru.
TIF(F)	Tagged Image File Format. Jeden z nejstarších a dodnes nejrozšířenějších formátů pro zpracování digitální bitmapové grafiky. Stejně jako u většiny formátů má i TIFF několik verzí. Jedna z nich umožňuje ukládat data i s tzv. bezztrátovou kompresí dat (viz níže), jiná ukládá data zcela nekomprimovaná.
JP(E)G	Soubory ve formátu JPEG (správně JPEG Interchange Format – JFIF; samotná zkratka JPEG je akronym z Joint Photographic Experts Group, tj. označuje skupinu, která stála za jeho vytvořením, nikoliv název formátu) jsou dnes asi nejrozšířenějším standardem bitmapové grafiky. JPEG je založen na tzv. ztrátové kompresi, při které dochází k jisté deformaci obrazových dat, čemuž se budeme blíže věnovat v následující podkapitole.

Abychom byli zcela přesní, je třeba dodat, že většina zde uvedených formátů existuje kvůli historickému vývoji či jiným souvislostem v několika variantách. Nejmarkantnější je tento fakt u formátu TIFF či JPEG (kde existuje i varianta JPEG2000 s lepší kompresí).

3.2.4 Problematika komprese

Fakt, že bitmapová grafika vytváří datově velké soubory, mnohdy v řádu desítek či stovek megabajtů (MB; pro srovnání CD médium mívá cca 720 MB, DVD 4400 MB) informací, vedla k vytvoření formátů schopných komprese dat.

Starší variantou je tzv. bezztrátová komprese, tj. takové zpracování grafiky, ve kterém nedochází ke ztrátě žádné informace či kvality obrázku. Obecně řečeno jde o takový kompresní algoritmus dat, který umožňuje stoprocentní zpětnou rekonstrukci těchto dat. Tato komprese se dá používat na všechny možné druhy informací – audio, video či textová data a mnozí uživatelé se určitě setkali s programy, resp. formáty, které byly vyvinuty speciálně pro tento druh komprese, jako je ZIP, RAR či méně známé gzip nebo LZMA.

Máme-li uvést příklad na fungování bezztrátové komprese, pak musíme maximálně zjednodušovat. Zkusme si představit výroční zprávu Univerzity Palackého. Je jasné, že v takovém textu se slovní spojení „Univerzita Palackého“ objeví mnohokrát, stejně jako třeba výrazy „fakulta“, „rektor“ či „student“. Program, který se bude snažit zkomprimovat daný textový soubor, pak může na základě některého z rozšířených algoritmů vyhodnotit, že při ukládání souboru na disk bude vhodné tato sousloví nahradit kratšími a vzájemné vztahy těchto nahrazených dat přiložit k souboru jako jakýsi dekomprimační klíč. Tak „Univerzita Palackého“ bude nahrazena jinak se nevyskytujícím výrazem „4p*“, „rektor“ složením „_!“ atd. Při takové náhradě dojde pak k úspoře místa, byť relativně malé – může však jít o megabajty informací v případě, kdy půjde o velká obrazová data atd. Z popsaného „pří-

kladu", který je samozřejmě jen orientačním, zvláště co se komprese obrazových či audio dat týče, je také jasné, že při následném používání (otvírání) již uloženého datového souboru dochází k jeho zpětné dekomprimaci, kdy jsou substituovaná data uvedena do čitelné, kompletní podoby.

Mnohem složitější je ztrátová komprese, při které dochází (což asi není překvapením) k určité ztrátě dat. Nejčastěji se tento typ komprese používá při ukládání zvukových (populární MP3 formát) nebo obrazových/video dat; naopak není možné jej využít pro kompresi textu.

Opět obecně: při ztrátové kompresi jsou data vyhodnocena a rozříděna na důležitá a méně důležitá, která jsou následně alespoň částečně obětována. Důležitá data je pak dále možné komprimovat i bezztrátovou metodou. Vzhledem k tomu, že nás primárně zajímá problematika ztrátové komprese u bitmapové grafiky, omezíme se pouze na ni a na nejznámějšího zástupce tohoto typu komprese, formát JPEG (JFIF).

Komprese bitmapové grafiky funguje na jednom základním principu, tj. předpokladu, že lidské oko není schopné zaznamenat všechny drobné nuance digitálního obrazu, přesněji všechny barevné nuance. Jednotlivé pixely je tedy pak možné zbavit jejich originálních barevných vlastností a celý barevný model pro určitou skupinu pixelů („barevné kostky" 8×8 pixelů aj. známé na digitálních fotografiích s příliš velkou kompresí), resp. pro celý obrázek matematicky odvodit a dopočítat. Při převodu se vychází také z faktu, že lidské oko je citlivější na jasovou složku obrazu než na jeho barevnost, jas se proto ponechává jako primární komponenta obrazové informace.

JPEG formát je pro určitý druh dat schopen dosáhnout i poměrně zásadního kompresního poměru vůči původnímu datovému souboru, nejednou v hodnotách 1 : 50 i vyšších, a to při takové ztrátě barevné informace, kterou lze označit za zanedbatelnou. Obecně špatných výsledků by však tento princip „matení nepřítel", tj. uživatele-diváka dosáhl u obrázků s výrazně kontrastními plochami, celkově malým počtem barev, ostrými přechody atd. Je také potřeba si uvědomit, že při práci s obrazovými daty ve formátu JPEG dochází k neustálé úpravě barevného modelu a přepočítávání komprese a obrazových dat – každá operace typu natočení, převzorkování, vložení barevné informace atd. vede k lehce pozměněnému výsledku. Je sice pravděpodobné, že tento rozdíl lidské oko nepozná, především pak při nastavení vysoké kvality (= menší komprese) obrazu, při kumulaci velkého počtu zásahů do původního zdroje se však lehce může stát, že výsledné barvy budou nakonec přece jen lehce posunuté od originálu.

3.2.5 Rozlišení obrazu, DPI a LPI

Při určování kvality a/nebo velikosti bitmapové grafiky se často setkáme se zkratkou DPI, pocházející z anglického „dot(s) per inch", tedy česky „body na palec". Již z toho vyplývá, že jde o termín, který udává jakýsi počet pixelů na fixní měrnou jednotku, jíž je 1 palec (inch; česky také „coul"), tj. 2,54 cm – a míněn je je-

den palec na vstupním nebo častěji výstupním zařízení.

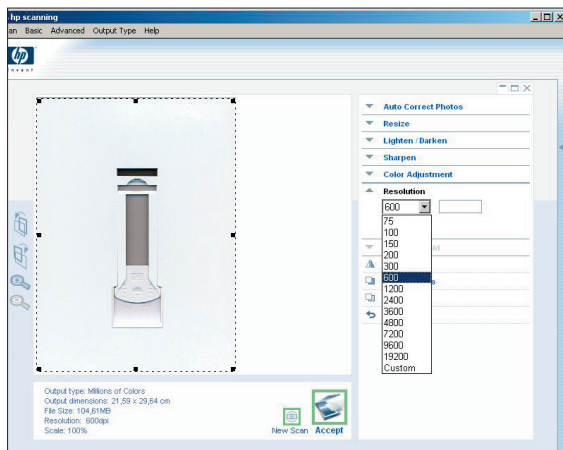
Vzhledem k tomu, že obrazová data jsou tvořena rastrem bodů, je zřejmé, že v případě získávání dat (skenování atd.) výše hodnoty DPI označuje, do jak husté mřížky je skenovaná předloha rozpoznávána, a tedy jak detailní bude popis získaného obrázku. Stejnou předlohu je samozřejmě možné neskenovat na libovolné DPI, ovšem jen relativně vysoká hodnota DPI

umožní další operace s takto získanou grafikou – jiná bude, pokud obrázek chceme vytisknout v rozměru 1:1 k předloze (pak se výše vstupního DPI bude zhruba rovnat hardwarovému rozlišení tiskárny atd.), jinou hodnotu DPI (vyšší) budeme však potřebovat, chceme-li neskenovaný obrázek dále pro výstup zvětšovat, pořizovat výřezy atd.

Jedná-li se o DPI uvedené ve vztahu k výstupnímu zařízení, tedy většinou tiskárně nebo při přípravě předtiskových předloh ve formátu PDF atd., pak toto číslo uvádí, jak kvalitní bude tiskový výstup, resp. kolik jednotlivých bodů dokáže tiskárna vytisknout na dané ploše (neboť obrázky, jak lze nejjednodušeji vidět u tištěných novin, jsou tvořeny rastrem bodů).

DPI ovšem není rovnocennou veličinou s vlastní velikostí obrázku v bodech. Jakákoliv bitmapa má svou velikost, udávanou v poměru „šířka“ × „výška“, tj. např. 1500 × 900 bodů. Je to ovšem hodnota nebo spíše nastavení DPI výstupního zařízení, co určuje velikost daného obrázku při tisku atd. Námi zmíněný obrázek, tištěný na tiskárně s nastavením 300 DPI, bude mít po tisku fyzickou velikost 1500 : 300 × 2,54 cm na šířku a 900 : 300 × 2,54 cm na výšku (tj. 12,7 × 7,62 cm; i laik pochopí, že jsem si modelové rozměry schválně dosti zjednodušil). V případě kdybychom stejný obrázek ale chtěli vytisknout na tiskárně s dvojnásobným rozlišením (600 DPI), dostaneme – dvakrát menší výstup.

Při „běžném“ tisku, kdy nedochází k zásadním deformacím předlohy, by mělo i ke kvalitnímu barevnému tisku stačit 300 DPI, byť i dnešní stolní tiskárny často nabízejí dvojnásobek. S ještě vyššími čísly operují výrobci stolních skenerů, tam jde

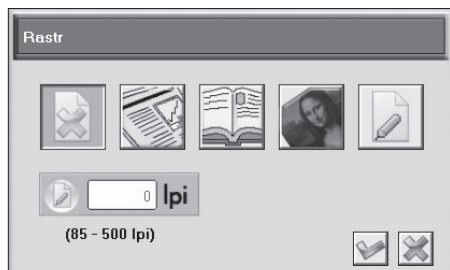


3.VIII Možnosti nastavení DPI při skenování předloh. Všimněme si nastavení palety barev (Millions of Colors, tj. 24bitová barva) a odhadu fyzické velikosti souboru před kompresí. Naskenovaný „obrázek“ je podavač pro skenování negativů a diapositivů, připojený na spodní stranu víka skeneru HP ScanJet.

však často o marketingový trik, kdy je hardwarová schopnost snímacího čipu násobena ještě interpolačním („dopočítávacím“) software. Nešvarem výrobců obecně je kromě udávání vyššího čísla také občasné zatajení faktu, že dané zařízení neumí skenovat stejně kvalitně v obou směrech, tj. má rozlišení např. 1200 × 600, je však vydáváno za očekávaných 1200 × 1200 DPI.

Méně „obvyklý“ termín LPI pak pochází z oblasti samotné polygrafie a označuje nastavení tiskového rastru, přesněji počet „linek na palec“ (line(s) per inch). Je to termín, který nás zajímá okrajově, neboť se týká již samotného procesu tisku, nicméně pro orientaci alespoň několik údajů. Hustota rastru při tisku na novinový

papír se pohybuje okolo 85 lpi, pro ofsetový papír je to nejčastěji 133 lpi a na natírané papíry to bývá mezi 150 a 200 lpi. Právě hustota rastru má svůj vliv na to, jak kvalitní mají být pořízené předlohy, tj. jaké by mělo být nastavení DPI při skenování atd. Obecně se uvádí přepočít 1–2krát (tzv. faktor kvality, fq) větší rozlišení obrázku, než je hustota rastru, závisí ovšem opět na tom, jak (fyzicky) veliký má být požadovaný výsledek.



3.IX: Možnost nastavení LPI u skeneru (zde Umax PowerScan - všimněme si zjednodušeného rozhraní, kde se automaticky nabízí nastavení LPI pro nejčastější problematické předlohy – noviny, časopisy atd.

Ačkoliv je LPI termínem z oblasti tisku, pro DTP pracoviště může být též důležitým v oblasti vstupních grafických dat – i u skenerů lze nastavit předpokládané (nebo samozřejmě známé) LPI, tentokrát ovšem u skenované předlohy. Kvůli tomu, že při tisku se používá pro vytváření výsledného obrazu určitý rastr vytištěných bodů, může se stát, že při nevhodném nastavení skeneru se na naskenovaném výsledku objeví nechtěný grafický prvek, známý jako moaré. Moaré efekt je samozřejmě obecně znám i z jiných oblastí jak běžného života, tak různých grafických a tiskových aplikací, důležité zde je však to, že většina skenerů nebo grafického software obsahuje nástroje, které tomuto nežádoucímu výsledku dokážou zabránit (viz obr. 3.IX).

3.2.6 Screen capture

Tzv. snímání obrazovky – screen capture – je metoda umožňující získat obrazová data toho, co se právě „odehrává“ na obrazovce počítače, ať již celé, nebo výřezu. Pomocí screen capture jsou mj. zachyceny a zde vysázeny všechny ukázky počítačových dialogů a různých nastavení, využití této funkce je ovšem širší a mnohostranné.

Základní metodu snímání obrazovky nabízí samotná Windows: stiskem kombinace kláves ALT a PrnScr (na některých klávesnicích označena nezkráceně jako Print Screen) se aktuální „snímek“ obrazovky, známý také jako screenshot, uloží do

schránky Windows, odkud je možné jej vložit do různých programů. Data se ve schránce ukládají jako poměrně nevhodná BMP bitmapa, je proto dobré obrázek před případným použitím v DTP oblasti uložit v jiném formátu. Pokud chceme udělat snímek ne celé obrazovky, ale pouze aktivního okna, resp. dialogu vybraného programu, můžeme výběr specifikovat pomocí kombinace CTRL + ALT + PrnScr.

Elegantnější a komfortnější variantou pro pořizování screenshotů je buď specializovaný software, např. CorelCAPTURE, nebo příslušné interní funkce různých grafických editorů.

Je nutné ještě jednou zdůraznit, že kvůli technickým parametrům monitorů, resp. fungování hardware vůbec, jsou výsledné obrázky snímány v malém rozlišení (běžné rozlišení monitorů je 96 DPI, je to ovšem orientační údaj; pro starší PC a Apple Macintosh platí 72 DPI) a jako takové vhodné spíše pro rozměrem menší tisk. Bylo-li by potřeba zvětšovat předlohu, která existuje pouze v digitální předloze malé velikosti, je tato metoda nevhodná (a velmi diskutovatelnou alternativou by pak byla metoda tisku a opětovného skenování na vyšší DPI).

4 BAREVNÉ MODELY A PRÁCE S BARVAMI

4.1 Barevné modely

Ve světě DTP se můžeme setkat s několika barevnými modely, tj. principy zobrazování či vytváření jednotlivých barevných odstínů na základě základních barev a jejich kombinací či vlastností. Jsou označovány zkratkami, z nichž mnohé na první pohled vypadají poněkud komplikovaně a tajemně, jako je RGB, CMYK, HSV nebo YUV. Detailněji si v této kapitole povíme něco o prvních dvou zmíněných a přidáme i několik dalších teoretických a praktických informací, které se k práci s barvami vážou.

4.1.1 Model RGB

Tzv. aditivní model míchání barev, označovaný zkratkou RGB, je vlastní všem zařízením, která barvu vytvářejí vyzařováním světla, jako jsou televizory, monitory počítačů, projektory, ale třeba také videohry či většina dnešních mobilních telefonů. Název tohoto modelu je odvozen od počátečních písmen tří základních barev, jichž je k míchání využito: červené – red, zelené – green a modré – blue. Aditivní se mu říká proto, že k černému základu se přidávají jednotlivé barevné složky, dokud není dosaženo bílé.

Hodnota jednotlivých barev se nejčastěji udává v absolutních číslech, může se nicméně zadat i v procentech či výjimečně v rozsahu 0 až 1. Pokud zůstaneme u obvyklého zápisu, pak budeme uvažovat o hodnotách 0–255 pro každou složku zvlášť, přičemž jejich krajní kombinace jsou tyto:

Red	Green	Blue	(výsledná barva)
0	0	0	černá
255	0	0	červená
0	255	0	zelená
0	0	255	modrá
255	255	0	žlutá
255	0	255	purpurová
0	255	255	azurová
255	255	255	bílá

Kombinací těchto barev, resp. jednotlivých odstínů těchto barev, je pak možné vytvořit plnohodnotnou barevnou škálu, tak jak ji známe z již výše zmíněných elektronických zařízení (odmyslíme si pro tuto chvíli případná hardwarová omezení).

RGB model je mj. základem pro nastavování barev při vytváření webových stránek, kde se používá buď v hexadecimálním zápisu (FF, 00, 00; tj. hodnota color="#ff0000" pro červenou barvu) nebo rovnou v zápise v RGB hodnotách (color="255,0,0"). (Při psaní rovnítká a jiných znaků v HTML kódu nedodržujeme standardní typografická pravidla, rovnítko tedy není oddělováno mezerou.)

4.1.2 Model CMYK

Tzv. subtraktivní model barev je vlastní všem tiskovým výstupním zařízením, tj. těm, kde se samotná barva vytváří odrazem, resp. lomem světla od povrchu nějakého předmětu. Je to tedy model, se kterým pracují všechny tiskárny a tiskaři světa, a který je pro DTP a pre-press základní. Subtraktivní proto, že jednotlivé barvy se „odečítají“ od bílé.

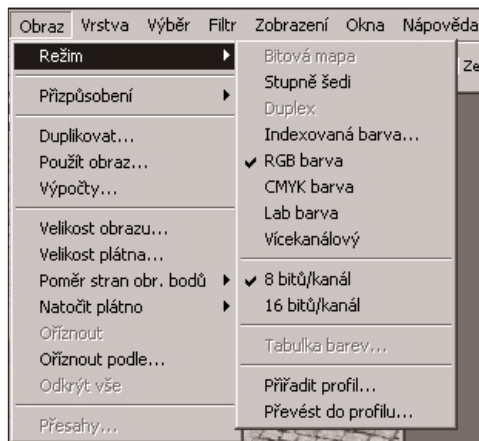
Stejně jako u předchozího modelu jde i v tomto případě o zkratku vytvořenou z počátečních písmen názvů některých barev, tentokrát cyan (azurová), magenta (purpurová), yellow (žlutá) a black (černá), respektive Key (tzv. klíčová). Hodnoty se udávají v procentech, přičemž model má některé styčné body s předcházejícím – např. hodnota CMYK 100, 0, 100, 0 dává zelenou atd.; mluvíme o systému komplementárních barev. Celý model má ovšem „otočené“ krajní hodnoty, tj. nulové hodnoty vracejí bílou barvu (předpokládáme samozřejmě vlastně nepotříštěný bílý papír), stoprocentní hodnoty výslednou černou.

U této černé/klíčové barvy je nicméně nutné se zastavit. Zatímco ryze teoreticky (fyzikálně) by mělo k vytvoření černé barvy stačit stoprocentní smíchání prvních tří složek, v praxi se zavedla čtvrtá, tzv. klíčová barva, reprezentovaná samotnou černou. Důvody k tomu jsou dva – 1. barvy používané při tisku při smíchání (z důvodů chemicko-technologických) dávají ve výsledku barvu nikoliv černou, ale spíše velmi tmavě šedou; 2. svou roli hraje ekonomické hledisko, neboť je vždy levnější tisknout jednou (černou) barvou, než k jejímu vytvoření používat jiné tři složky. Černá je navíc sama o sobě levnější než barvy s pigmentem.

(Existují však domácí, stolní tiskárny, většinou inkoustové, které fungují na principu CMY. Mnohdy nižší pořizovací náklady těchto přístrojů pak bohužel při užívání negativně vyváží cena tiskových náplní, a to i při „černobílém“ tisku, kdy je výsledná černá vytvářena smícháním všech tří barevných složek.)

Výsledný „barevný“ obrázek je pak tvořen soutiskem (překrytím) jednotlivých C-M-Y-K složek, tedy navrstvením barevných pigmentů jednotlivých barviv na sebe tak, aby vznikala plocha, která různým způsobem odráží světlo a vytváří tím barevné spektrum. I to je jeden z důvodů, proč v tiskovém modelu CMYK existuje samostatná černá – při tisku např. velmi malého černého písma či různých drobných detailů by byl soutisk jednotlivých složek někdy až příliš obtížný.

Jak je zřejmé z výše uvedeného, grafické prvky z modelu RGB je nutné převádět na model CMYK. To dělá, podle náročnosti a kvality tisku a výstupního zařízení, buď samotná tiskárna, nebo ripovací zařízení, nebo uživatel sám. K tomu slouží např. při práci s digitální fotografií možnost nastavení, v jakém barevném modelu se s nimi má pracovat (viz obr. 4.1; jde o nastavení ve Photoshopu). To, že samotný monitor počítače stále pracuje v modelu RGB, je samozřejmě jiná věc. DTP programy jsou většinou přednastavené pro model CMYK (může se v nich však připravovat třeba výstup do PDF nebo XML, určený primárně pro obrazovku, pak je nutné uvažovat o přechodu na model RGB).



4.1: Nastavení barevného modelu

fických řad nebo specializovaných výrobců, jako je Eizo, nabízejí celou řadu softwarových i hardwarových nástrojů k nastavení co nejvíce odpovídajícímu barvám výstupu), té je však obtížné dosáhnout, navíc v prostředí menších DTP pracovišť nebo jednotlivých uživatelů je i zbytečná. Barvy totiž samozřejmě není možné vybírat pomocí monitoru – nebo jen velmi přibližně –, ale pomocí speciálních vzorníků barev. Ty jsou buď mezinárodní a široce platné (Pantone process colors aj.), nebo specifické a ověřené pro jednotlivá grafická studia, respektive tiskárny a tisková zařízení.

Ještě jedno praktické upozornění: samotné krajní hodnoty, tj. 0–0–0–0, respektive 100–100–100–100, jsou v praxi někdy nepoužitelné, především při tisku barevných fotografií, neboť na bílých místech vytváří pocit „vypáleného“ místa, až nepatřičně vyběleného. Tam, kde jsou fotografie opravdu bílé, se při tisku fakticky potiskují malou hodnotou šedi nebo jiných odstínů, tak aby lidské oko vnímalo tuto část fotografie odlišně od bílého papíru. Stejně tak překrývání stoprocentních hodnot jednotlivých složek je z různých praktických důvodů často nepoužitelné či nepoužívané (barvy špatně schnou, rozpíjí se atd.).

4.1.3 Jiné modely pro práci s barvami

Dalším možným modelem je kombinace HSV, vycházející z označení Hue (barevný tón), Saturation (sytnost barvy) a Value (hodnota jasu); někdy paralelně označovaná HSB (Brightness místo Value). Udává se v různých hodnotách: zatímco odstín ve stupních v hodnotách 0–360 (vyjadřuje polohu barvy na modelu, který je zobrazen jako barevné kolo), sytnost a jas v procentech. Je to model, který reálně nejlépe odpovídá vnímání lidského oka; technicky má blízko k modelu RGB. I s tímto modelem je možné se setkat v grafických aplikacích.

Model Lab (L jako luminance, tj. světlost, zatímco ostatní dvě položky udávají hodnoty dvou barevných složek) je mezinárodně uznávaným (vytvořen komisí Com-

Je nicméně nutné si uvědomit, že ačkoliv monitor zobrazuje vždy barvy v modelu RGB a dokáže „pouze“ emulovat tiskový model CMYK, vždy bude existovat rozdíl mezi barvami, tak jak je vidíme na monitoru (obzvláště na negrafických, levných LCD) a jak budou nakonec vytištěny, ať již na stolní laserové tiskárně, nebo na profesionálním čtyřbarevném ofsetovém tisku. Tento rozpor se dá částečně odbourat přesnou kalibrací používaných monitorů (monitory vyšších gra-

mission Internationale d'Eclairage, známý proto také jako CIE Lab) modelem pro definování barev. Jeho výhoda je v tom, že je nezávislý na zařízení, vytváří tedy shodné barvy jak např. na monitoru, tak v tiskárně atd. I tento model se používá v grafických aplikacích, třeba ve Photoshopu při operacích s barevnou bitmapovou grafikou dokonce interně.

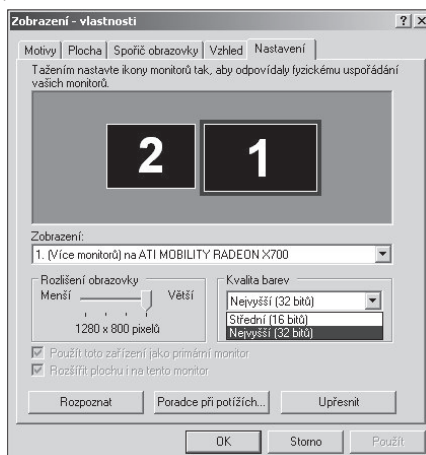
Model YUV je model kódování televizního signálu (NTSC, PAL, SECAM systémy), má tedy opět blíže k modelu RGB. Jednotlivé znaky označují jas (Y), resp. dvě barevné složky (U, V).

4.2 Barevná hloubka

Barevná hloubka – termín spojovaný především s bitmapovou grafikou, ale jinak platný obecně a vztahující se i k hardware počítače – definuje počet barev, které je každý pixel schopen zobrazit. Uvažujeme-li o barevné hloubce, pak musíme brát v úvahu také vstupní zařízení, tj. skenery, fotoaparáty atd., kde patří nastavení barevné hloubky k jednomu ze zásadních vstupních údajů.

Barevná hloubka udává, na jakou šíři/paletu barev je nastavené dané zařízení, resp. pořizovaný či existující obrázek. Vychází se přitom a) z minimálního zobrazitelného počtu barev – 2^1 (tedy černá a bílá) a následně mocnin tohoto základu; b) z faktu, že ne u každého obrázku je nutné nastavit maximální možnou barevnou hloubku, resp. že ne všechna digitální zařízení jsou schopná zobrazit větší počty barev (nostalgičtí si jistě vzpomenou na grafické karty zobrazující jen 4 nebo 16 barev). Je samozřejmé, že vyšší barevná hloubka s sebou přináší fyzicky větší soubory bitmapové grafiky, co se velikosti v bajtech týče.

Nevýhodou při udávání barevné hloubky je jistá nejednotnost jejího popisu. Udává se buď v absolutních číslech (zmiňované 4,16 nebo 256, ale také dnes běžných 16,7 milionu barev atd.) či v bitových řádech, kde poslednímu zde uvedenému absolutnímu údaji odpovídá „24bitová hloubka“ nebo podobně v počtu bitů na kanál (kanály vycházejí z barevného modelu, RGB model má tedy tři kanály; na obrázku 4.1 je pak vidět nastavení pro „8bitový RGB“ obrázek). Používá se také anglické označení „High Color“, resp. „True Color“, v češtině se někdy při nastavování kvality zobrazení setkáme s nešťastným relativním popisováním typu „Střední kvalita“ nebo „Nejvyšší kvalita“ (platí pro nastavení barevné hloubky grafických



4.II: Nastavení barevné hloubky, se kterou bude pracovat příslušný monitor

karet/monitoru, viz obr. 4.II). Mnozí uživatelé znají také označení „EGA“, „VGA“ a podobné, která se nicméně vážou primárně k hardware počítače.

Z praktického a historického hlediska je většina jak formátů, tak zařízení omezená na 24bitovou hloubku. Toto omezení je možné kompenzovat softwarovou korekcí, běžnou např. už i u levnějších skenerů nebo fotoaparátů, ideálem pro nejkvalitnější využití je nicméně 48bitový TIFF formát, se kterým však pracuje málokdo a máloco.

Pro přehlednou orientaci:

počet bitů	mocnina	počet barev
1bitová barva	2^1	2
2bitová barva	2^2	4 barvy (CGA)
4bitová barva	2^4	16 (EGA)
8bitová barva	2^8	256 (VGA), typická také pro obrázky ve stupních šedi (monochromatický 8bit)
16bitová barva	2^{16}	65 536 (High Color)
24bitová barva	2^{24}	16 777 216 (True Color)
32bitová barva	2^{32}	teoreticky 4 294 967 296 (True Color)

4.3 Gamut barev

Gamut barev je rozsah barev, který je dané výstupní zařízení schopné zobrazit, a to nezávisle na modelu. Jiný gamut mají např. různé tiskárny (či spíše tiskové technologie), jiný gamut mají monitory, LCD atd. Pokud dané zařízení není schopné zobrazit některé barvy, jsou tyto nahrazeny jinými, přibližně odpovídajícími (nemělo by tedy dojít k jakési úplné ztrátě barevné informace).

Gamut modelu RGB je větší než u tiskového modelu CMYK, může se proto stát, že některé barvy, které vidíme dobře zobrazené na monitoru, se nepodaří stejně kvalitně vytisknout (mezi tyto barvy patří i např. krajní barvy modelu RGB, tedy „čistá“ červená atd.). Převod mezi modely může vést k lehkým barevným deformacím, nedoporučuje se proto například opakované převádění mezi modely RGB a CMYK atd.

4.4 Přímé barvy, duplex

Pro spoustu tisků, které jsou svým způsobem „barevné“, není třeba využívat principu CMYKového tisku, tedy soutisku čtyř barevných vrstev. Mnohdy je možné barevného výsledku docílit kombinací černá + doplňková barva. Tyto doplňkové barvy se nazývají přímé barvy a jsou většinou označovány mezinárodně uznávanými kódy a míchány separátně. K jejich výběru slouží barevné vzorníky jednotlivých výrobců, z nichž nejspíše nejznámější a nejrozšířenější jsou vzorníky systému Pantone (Pantone spot colors; je nutné rozlišovat od Pantone process colors, což

jsou barvy tištěné pomocí klasické CMYK metody, viz výše; existují samozřejmě i jiné vzorníky, např. Truematch).

Tato metoda se používá nejčastěji tam, kde je použita jen jedna barva, je však nutné naprosto přesně uchovat její barevné kvality, tedy např. při tisku firemních log atd.

Přímé barvy se využívají také v kombinaci s celou CMYKovou škálou v případě, že požadovaný obrázek obsahuje barvu, kterou nelze získat samotnou kombinací složek C-M-Y-K, tj. např. metalické, neónové (fluorescenční) a jiné speciální barvy. Tato přidaná přímá barva (případně i dvě) se pak tiskne jako 5. (a 6.) složka tiskového procesu (což mnohdy znamená, že papír prochází tiskařským strojem dvakrát – pokud není možné přidat ke 4 základním plátům při tisku žádný další).

Duplexní tisk (dvojtónový) označuje technologii, kdy je ve výsledku původní obrázek ještě doplněn další barvou. Nejjednodušeji si lze tuto technologii představit na černobílých fotografiích. Zatímco k některým pro punc starobylosti je při tisku často přidáván odstín okrové („sépíe“), jiné mohou dostat třeba naopak lehký odstín modré pro zdůraznění modernosti či technického prvku atd. Kvalitní černobílé umělecké fotografie se obecně většinou tisknou v duplexu černá – šedá. Podklady pro duplexní tisk se připravují softwarovým zpracováním předlohy, nejčastěji ve Photoshopu a podobných programech.

5 PÍSMO

5.1 Písmo jako předmět typografie

Jistým paradoxem je, že pokud se DTP rozvíjí v posledních deseti či patnácti letech, většina klasických písem má za sebou mnohdy více než staletou historii. Písma jako Baskerville, Bodoni nebo Garamond zde byla mnohem a mnohem dříve, než se ve čtyřicátých letech 20. století začalo mluvit o prvních prakticky využitelných počítačích. S tím souvisí i fakt, že mnohá pravidla, vlastnosti a termíny, které se dříve používaly v typografii a sazbě, mají svůj původ mimo počítačové programy, se kterými si ne vždy ideálně rozumí.

Typografie je samozřejmě obor s velkou minulostí i budoucností a jeho pole se jen částečně – ovšem zásadně – dotýká problematiky počítačové sazby. V této a následující kapitole se seznámíme jen se základními pojmy a vlastnostmi, pro hlubší vhléd je nezbytné ponořit se do některých učebnic a publikací uvedených v přehledu literatury.

5.1.1 Řez a rodina

Řez písma a rodina písma jsou termíny, které spolu úzce souvisejí, neboť rodina písma je definována jako množina všech řezů daného písma. Řezem pak rozumíme kresebné varianty všech znaků daného (typu) písma, používané v praxi nejčastěji k vyznačování některých úseků textu – mezi obecně známé řezy patří kurzivní řez („kurzíva“), tučný řez a jejich kombinace – tučná kurzíva. Výchozím je samozřejmě základní řez, tj. základní písmo (nazývané v závislosti na terminologii také regular či roman).

Plně vytvořená rodina písma by však měla obsahovat více řezů než jen základní set, protože pro vyznačování slov a úseků textu i pro vlastní sazbu je někdy vhodné sáhnout i po jiných variantách písma – proto máme řezy polotučné, extra tučné, zúžené nebo naopak odlehčené, slabé. (Anglická terminologie je v tomto směru bohužel nejednotná, objevují se varianty demibold/semibold; extra bold nebo extra heavy, ale i mnohé další.) Řezy se tedy liší ve třech základních vlastnostech: tloušťce (duktu), znakové šířce či sklonu znaků. Kompletní rodina písma může obsahovat serifové i neserifové varianty jednoho písma.

Tloušťka písma se někdy ztotožňuje s duktem písma. Duktus definujeme jako poměr tloušťky písmových tahů k výšce písma, rozeznáváme tedy písma s duktem základním, zeslabeným a zesíleným.

V jedné sazbě není pro zvýrazňování textu vhodné kombinovat polotučné a tučné řezy stejného písma. Samotné polotučné písmo, ač je v posledních letech vinou zjednodušujících textových procesorů vytlačováno, působí překvapivě často estetičtěji než tučné řezy. Je nutné si dávat pozor také na tzv. falešnou kurzívu – mnohé programy tam, kde používanému písmu chybí vytvořený kurzivní řez, vytvářejí na přání uživatele „kurzívu“ tak, že jednotlivé znaky písma jednoduše nakloní v požadovaném úhlu (kolem 10 stupňů). Kurzíva ovšem není jen „nakloněným“ základním řezem písma, ale samostatnou sadou glyfů (znaků), samostatným řezem

písma! Nejčastěji poznáme falešnou kurzívu pomocí špatně vykresleného znaku „a“, „k“, „z“ a jiných, srov.:

normální text	normální text
<i>kurzíva</i>	<i>kurzíva</i>
<i>falešná kurzíva</i>	<i>falešná kurzíva</i>
(Futura)	(Palatino)

Výraz „rodina písem“ se někdy chápe trochu širěji jako skupina písem s podobnými nebo stejnými charakteristikami. Je to terminologie, která se ujala při vytváření webových stránek (přesněji při vytváření tzv. kaskádových stylů, CSS) a která pomáhá tvůrcům a designérům webu nařídit počítači, s jakými fonty a při jaké prioritě má pracovat při zobrazení určitého úseku textu. Definice rodiny podobných písem pomocí CSS by pak mohla vypadat například takto: font-family : Arial, Helvetica, sans-serif.

5.1.2 Verzálky, kapitálky, minusky...

Jedná se o kategorie znaků: verzálky nebo majuskule jsou správným typografickým názvem pro velká písmena, minusky nebo minuskule pro písmena malá. Kapitálky označují verzálky zmenšené na tzv. střední výšku písma (viz nákres vykreslování písma, 5.1).

I u kapitálek platí, že rozlišujeme tzv. pravé kapitálky, což je vlastně další řez daného písma nebo skupina přesně definovaných glyfů v jednom písmu, a kapitálky elektronické, které počítač automaticky vytváří snížením verzálek podle předem nastavené hodnoty.

Ačkoliv se na to mnohdy zapomíná, rozlišují se také minuskové a verzálkové číslice. Zatímco počítačové programy většinou pracují s verzálkovou sadou číslic, které jsou stejně „vysoké“ jako verzálky, minuskové číslice jsou definovány různě: číslice 1, 2, 0 jsou vykreslené na střední výšku písma, ostatní číslice tuto výšku přesahují buď jedním nebo druhým směrem.

5.1.3 Kresba a konstrukce písma

Při popisu jednotlivých znaků i celého písma vycházíme jednak z tzv. písmové osnovy, což je soustava souřadnic, do kterých je každé písmo ukotveno, jednak z jednotlivých tahů, ze kterých se skládají vlastní písmena. Základní orientaci v písmové osnově nabízí obrázek 5.1.



5.1: Písmová osnova

Přesnou typografickou terminologii můžete nalézt jednak v oborové normě ČSN 88 0111, jednak opět v příslušné literatuře nebo i na internetu. Doporučujeme alespoň základní orientaci v termínech jako jsou dřík písmena, bříško, oko, příčka atd.,

samozejmě pak seznámení se s jedním ze základních typologických prvků písma, serifem neboli patkou.

Sířední výška písma se označuje také „x-height“, neboť odpovídá velikosti litery „x“.

5.1.4 Velikost písma

Určit přesně velikost písma, respektive shodnout se na tom, jak se bude velikost písma a z ní vycházející vlastnosti sazby definovat, je někdy překvapivě komplikované. Je to opět důsledkem faktu, že základní měrný systém, tzv. Didotův, o staletí předešel dobu elektronické sazby. Vše je navíc komplikováno faktem, že další z používaných systémů, upřednostňovaný v DTP, přišel ze zámoří, není tedy jednoduše převoditelný například na metrický měrný systém, jak bychom si nejspíše přáli.

5.1.4.1 Didotův měrný systém

Měrný systém Francouze F.-A. Didota, pocházející z 80. let 18. století, se stal na staletí standardem pro klasickou sazbu. Jeho základem je typografický bod, dvanáct bodů pak tvoří jedno cicero. Převod na metrický systém je mírně komplikovaný:

1 bod = 0,375 mm (jako jeden didot platný ve standardech EU)

12 bodů = 1 cicero = 4,513 mm

2 600 bodů = 1 m

Didotův měrný systém je dnes v počítačové sazbě již téměř opuštěn ve prospěch dalšího měrného systému, kterým je měrný systém point/pica. Po rodině Didotů, francouzských tiskařů, vydavatelů i typografů, přesněji po jeho tvůrci Firminu Didotovi (1746–1836) je také pojmenováno jedno ze známých písem, které je dnes v nabídce mnoha písmolijen po celém světě.

5.1.4.2 Měrný systém point/pica

Americký měrný systém pica [pajka], vytvořený jako americký celonárodní standard v roce 1886, má sice jako základní jednotku také bod (říká se mu raději point, pt), ten má však jinou velikost: 0,3528 mm. Dvanáct bodů tvoří jednu pica, šest pica jeden palec (inch). V číslech přehledněji ještě jednou:

1 pt = 0,3528 mm

12 pt = 1 pica = 4,23 mm

72 pt = 6 picas = 1 palec = 2,54 mm

American Point System, jak zní oficiální název tohoto měrného standardu (někdy je nazýván také monotypovým; monotype), měl ve skutečnosti složitou genezi a velikost jednoho pointu neodpovídá poměru uvedenému výše. To je způsobeno tím, že po nástupu elektronického DTP byl pro sázecí programy tento měrný systém lehce upraven pro lepší manipulovatelnost. Z toho také vyplývá, že právě s upraveným měrným systémem point/pica se setkáváme při práci s počítačovou grafikou.

5.1.4.3 Kuželka & Čtverčík

Kuželka je reliktem kovové sazby, kdy označovala hranolek s písmem. Výška kuželky ve strojové sazbě byla součtem stupně písma a řádkového prokladu, odstupu jednotlivých řádků, který je připojen pod samotnou literu na kuželce; ve starší ruční sazbě byla kuželka rovna stupni písma, proklad se nastavoval nezávisle. Akcenty písem mohou přesáhnout kuželku, proto při sazbě akcentovaného písma, což je případ češtiny, může dojít k nepříjemnému překrývání řádků. Zjednodušeně a nepřesně se někdy velikost kuželky ztotožňuje s velikostí písma.

Z kuželky se odvozuje čtverčík, což je rozměr pomyslného čtverce o délce stran odpovídající velikosti kuželky písma. Pro zjednodušení lze říci, že čtverčík odpovídá přibližně velikosti písmene „M“. Čtverčík, a to je důležité, je tedy jednotka s relativní hodnotou, která se odvíjí od velikosti písma – Times 10 pt má čtverčík 10×10 pt atd.

Pokud kuželka je dnes spíše abstraktním reliktem typografie, čtverčík stále zůstává jedním z nejzákladnějších prvků při definici některých rozměrů a při sazbě samotné. Podle čtverčíku totiž definujeme velikost některých dalších prvků (např. odstavcové zarážky, o které bude zmínka v následující kapitole), především různých typů mezer. V sazbě totiž rozlišujeme více než obyčejný „mezerník“ – mezery se charakterizují především dle jejich poměru ke čtverčíku, můžeme se tedy setkat s mezerou čtverčíkovou (podle anglické terminologie *em space*, tedy česky někdy slangově *emková mezer*a), půlčtverčíkovou (*en space*; „normální“ mezer)a; osminkovou nebo vlasovou (*hair space*), což je $\frac{1}{24}$ čtverčíku. Všechny tyto mezery mají navíc pevnou velikost, vypočítávanou podle čtverčíku písma, na rozdíl od mezery „normální“, vkládané pomocí mezerníku, jejíž velikost si program dle potřeby v určitém rozmezí přizpůsobí. Také tzv. nerozdělitelná nebo pevná mezer, která slouží ke spojování dvou slov nebo znaků, má proměnnou velikost. Specifickou mezerou je také tzv. číslicová mezer neboli mezer a o šířce číslice, jejíž velikost se vypočítává dle daného písma, ale pak je neměnná. Používá se pro sazbu číslic do sloupců, je-li potřeba tato čísla nějakým způsobem zarovnat. Kompletní přehled různých typů mezer přináší jakákoliv typografická příručka, ale naštěstí pro nás je jejich výběr a vkládání např. v InDesignu intuitivní – přes menu *Formát* a položku *Vložit* prázdné místo si vybereme, kterou mezeru na chceme umístit na místo kurzoru. Ty nejčastější mají také své předdefinované klávesové zkratky:

půlčtverčíková (<i>en space</i>) mezer:	CTRL + Shift + N
osminková mezer:	CTRL + ALT + Shift + M
vlasová mezer:	CTRL + ALT + Shift + I
číslcová mezer:	CTRL + ALT + Shift + 8
nerozdělitelná mezer:	CTRL + ALT + X

5.1.4.4 Tradiční udávání velikosti písma

Kromě určování velikosti, resp. stupně písma, ve výše určených měrných jednotkách existuje také tradiční pojmenování pro určité velikosti používaného písma.

Uvedme si alespoň některé názvy a hodnoty: diamant (4 b), kolonel (7), petit (8), borgis (9), garmond (10), cicero (12), střední (14), tercie (16), text (20). Tato klasifikace je jednoznačně na ústupu.

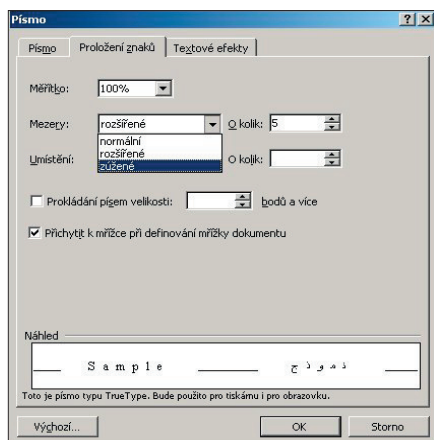
5.1.5 Kerning a prostrkání

Kerningem, česky vyrovnáváním, se chápá úprava mezer mezi některými dvojicemi znaků, jejichž kuželky se při elektronické sazbě opticky „překrývají“. Jako příklad si můžeme vzít majuskule „A“ a „V“. Pokud by program nesáhl k optickému překrytí těchto jednotlivých glyfů, byla by mezi nimi příliš velká mezera (viz obr. 5.II – nahoře teoretická ukázka sazby bez kerningu, dole se zapnutým a funkčním kerningem). Naopak u některých kombinací písem („r“ a „n“ aj.) by hrozilo, že dojde k jejich optickému spojování, a tím vzniku nechtěných slitků. I to by měl správně nastavený kerning ošetřit. Faktem bohužel je, že ne každé (počítačové) písmo obsahuje všechny potřebné informace o vyrovnávání vybraných dvojic znaků, může proto při použití v DTP programech docházet k optickým chybám.

Určitou variantou ke kerningu jsou tzv. slitky (ligatury), což jsou předem definovaná spojení dvou znaků, např. Æ. Větší možnosti ligatur nabízejí především moderní OpenType písma (viz níže).

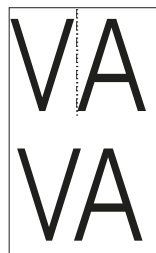
Pravděpodobně nejdůležitější je nastavení nebo spíše ruční úprava vyrovnání párů při sazbě verzálkami nebo kapitálkami ve velkých velikostech písma. Kerning není totéž co prostrkání nebo rozpal písma, ačkoliv jde o podobné typografické prvky!

Prostrkáním rozumíme ruční úpravu vzdáleností mezi písmeny, která může vést k tzv. volnější nebo naopak těsnější sazbě. Písmo se nesmí prostrkávat pomocí mezer, ale pomocí specifického nastavení písma! To platí jak pro sázecí programy, tak třeba i pro procesory, jako je MS Word, kde je prostrkání schováno pod vlastnosti písma, kartu Prostrkání znaků a položku Mezery (viz obr. 5.III), udávanou navíc v ne zrovna typografických jednotkách (prostrkání se jinak definuje v tisících

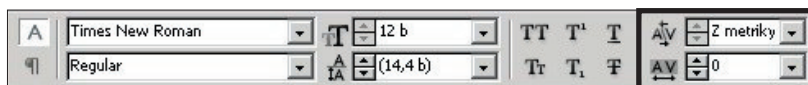


5.III: Prostrkání písma ve Wordu

nách čtverčíku s obvyklými hodnotami kolem 150–200). InDesign pro funkce vyrovnání, resp. prostrkání, nabízí na paletě Znaků dvě snadno přístupná nastavení hned pod sebou (obr. 5.IV).



5.II: Kerning



5.IV: Funkce vyrovnání a prostrkání v InDesignu (vpravo na panelu)

5.1.6 Klasifikace písem

Klasifikace jednotlivých písem bohužel (ovšem asi nepřekvapivě) není jednotná. Ve světě se používá několik různých metod, většinou spojených s některými firmami nebo vývojovými fázemi oboru (klasifikace IBM, Pantone aj.). Celosvětově rozšířená je také klasifikace Mezinárodní typografické asociace (Association Typographique Internationale, ATypI).

5.1.6.1 Solperova klasifikace

V České republice se používá klasifikace písem vytvořená Janem Solperou, která byla dříve také platnou ČSN (88 1101). Její charakteristiku shrnuje např. server Typo.cz takto: „Má řadu předností – snad největší v tom, že sice přihlíží k chronologickému vývoji písma, ale hlavní pozornost je soustředěna na grafické znaky písma. To umožňuje zařadit nejen písma existující, ale i ta, co teprve vzniknou.

Dělení písem odpovídá historickému vývoji od dynamického (rukopisného) principu k statickému (psanému) principu. Mezi nimi je vždy přechodová forma a tyto tři skupiny doplňuje čtvrtá, kde převládají smíšené prvky.

Do tří etap se dělí vývoj antikvových (patkových) písem. Dělení je aplikováno i na bezserifová a psaná písma, tam ale neodpovídá přesně chronologickému vývoji a řídí se podle grafických znaků. Čtvrtá skupina neodpovídá žádné etapě vývoje.“

Solpera klasifikuje tyto skupiny písem:

1. Dynamická antikva
2. Přechodová antikva
3. Statická antikva
4. Lineární písmo serifové
5. Lineární bezserifové statické písmo
6. Lineární bezserifové geometricky konstruované písmo
7. Lineární bezserifové dynamické písmo
8. Lineární antikva
9. Kaligrafická písma
10. Volně psaná písma
11. Písma lomená

5.1.6.2 Zjednodušená klasifikace podle druhů písem

Pro naše „školní“ potřeby je možné si jednotlivé skupiny písem zredukovat tím, že za základní *druhy* písem budeme chápat písma (1.) serifová a (2.) bezserifová (patková a bezpatková). U těchto skupin vnímáme jejich základní kresebný rozdíl, totiž (ne)existenci serifů – patek, a s ním spojené funkční rozdíly. Zatímco serifová písma se využívají pro delší texty, protože patky umožňují čtenáři lépe vnímat

kontinualitu textu, písma bezserifová jsou častější pro sazbu akcidenčních tiskovin (plakáty, vizitky) nebo pro odlišné prvky u hladké sazby jako jsou nadpisy kapitol, titulky atd. Pro beletrii a publicistiku se až na výjimky používají serifová písma, bezserifová jsou u knižní sazby častější u technických a odborných textů.

K této dvojici termínů je nutné připočítat několik dalších specifických skupin. Písma (3.) kaligrafická neboli skriptová jsou taková, jejichž kresba napodobuje psaní rukou. Písma (4.) lomená se vyznačují obecně velkými rozdíly mezi tloušťkou vertikálních a horizontálních tahů (dříky, příčky), jako typická varianta se nabízí německý švabach. (5.) Akcidenční písma používáme spíše výjimečně pro specifické účely – často napodobují dobová písma (art deco, historizující písma, futuristická písma). Konečně existuje také skupina (6.) ostatních písem, kam patří různé spíše obrázkové fonty nebo písma vytvořená pro specifické účely, jako byl již zmiňovaný fonetický přepis atd.

Tato pomyslná klasifikace nenahrazuje žádným způsobem ani Solperovu, ani jinou šířeji uznávanou klasifikaci písem!

5.1.6.3 Proporcionální a neproporcionální písma

Proporcionální písma jsou taková, jejichž znaková sada obsahuje glyfy s různou šířkou – tj. vlastně naprostá většina všech písem, neboť je logické, že písmena „i“ a „W“ mají různou šířku. Oproti tomu existují také i neproporcionální písma, zvaná též anglicky monospaced (což je termín asi lépe vystihující jejich podstatu). V takových písmech jsou všechny glyfy vytvářeny na stejnou šířku největšího možného písmene, tedy „M“. Je to písmo, které svým způsobem napodobuje psaní na psacích strojích a jako takové je až nečekaně často používané, především díky písmu Courier, které patří mezi systémová písma Windows. Ačkoliv neproporcionální písma mohou mít své specifické využití v typografii (tisk matric aj.), je naprosto nevhodné k nim sahat z jiných důvodů, např. kvůli snaze vytvořit „zarovnání do tabulky“ atd.

Často také vidíme využívání neproporcionálního písma tam, kde se autor z nevědomosti snaží dosáhnout psaní podle tzv. normostran, tedy rámce 60 znaků na 30 řádků. Normostrana je ovšem pouze pomocným, teoretickým měřítkem – rozsah textu jednoduše spočítáme pomocí interních programových funkcí pro kalkulaci počtu znaků (kteréžto číslo dělíme 1800).

5.2 Písmo jako předmět počítačové techniky

Z technického či počítačového hlediska je písmo vlastně vektorovou grafikou. Existují i bitmapová písma, ta si však pro naše potřeby můžeme odmyslet, protože nejsou použitelná v typografii. Součástí definice jednotlivých písem (fontů) však nejsou jen tvary jednotlivých znaků (glyfů), ale také definice jejich vlastností, vztahů, možností ligatur (sliktů) a mnohé další prvky.

Stejně jako může být samotná vektorová grafika uložena v různých formátech, existují i různé formáty písem, z nichž musíme rozlišovat formáty True Type, Type 1 a OpenType.

5.2.2 Formáty písem (fontů)

5.2.2.1 TrueType

Písma ve formátu TrueType (TrueType Font, proto zkratka a koncovka TTF) jsou pravděpodobně nejrozšířenějším formátem počítačových písem na světě. Tento formát, který vznikl jako konkurence Type 1 písem firmy Adobe (hlavně kvůli cenové politice), se stal základem pro práci nejen v systému Windows (od verze 3.1), ale i v appleovském systému Mac OS.

Dnes v tomto formátu existují desítky – ne-li stovky – tisíc různých písem a jejich jednotlivých řezů, byť samozřejmě jen zlomek z nich byl upraven pro použití ve středoevropských jazycích a/nebo speciálně pro českou abecedu s jejím relativně velkým množstvím diakritických znamének (akcentovaných znaků).

Písma ve formátu TrueType mívají bohužel mnohé nešvary. Maximální znaková sada, kterou je možné v rámci jednoho písma (řezu) definovat, je limitována 256 znaky, což je přece jen málo, ačkoliv se to tak na první pohled nemusí jevit. Je nutné si ale uvědomit, že písmo není jen základní sada písmen v majuskulích a minuskulích, ale také spousta speciálních liter, často specifických pro jednotlivé jazyky (ç, ů nebo ř atd.), a spousta jiných, pomocných glyfů (Δ, †, ©). Navíc mnohá písma vytvořená pro specifické účely či z důvodů dekorativních (tj. mnohá písma okázalá či lomená atd.) mnohdy nemají definovanou ani celou sadu, ale spojují se jen s několika desítkami základních znaků.

Druhá nevýhoda TrueType písem je fakt, že ne každé písmo je vytvořeno dostatečně precizně a kvalitně, aby bylo použitelné pro profesionální tisk: některá písma mají špatné proporce, některé křivky nejsou dobře definované, prostor mezi literami či postavení diakritických či interpunkčních znamének může být špatně nastavené atd. Z těchto důvodů (i z některých technických, např. špatná spolupráce s postskriptem) se mnohdy nedoporučuje používat TrueType fonty pro profesionální, kvalitní tisk a některá grafická studia s nimi apriori nepracují.

Na druhou stranu právě v TrueType formátu lze získat celou řadu velmi specifických fontů, z nichž mnohé mají užití především na akademické půdě či ve specializovaných, odborných tiscích – takové jsou fonty IPA pro fonetický přepis textu, zčásti volně šiřitelná písma pro sazbu hlaholice nebo cyrilice i jiné specializované fonty.

5.2.2.2 Type 1

Type 1 je vlastně nejstarší a základní formou postskriptového písma (zavedené u nejstaršího Level 1 PS již v 80. letech). Vzhledem k tomu, jak již víme, že pro zobrazení postskriptu jsou zapotřebí speciální programy, resp. zařízení (ripování postskriptu), bylo i pro zobrazení Type 1 písem ve starších verzích Windows a jiných systémů zapotřebí speciálního programu. Tím je Adobe Type Manager, podle kterého se těmto písmům někdy slangově říká také ATM fonty; další označení jsou

pak PostScript Type 1 nebo Adobe Type 1 atd. Windows od verze 2000 výše již žádný speciální program pro zobrazení Type 1 písem nepotřebují.

Type 1 fonty mají sice také nepraktické omezení na maximálně možnou množinu 256 znaků, ve většině případů však jde o profesionálně vytvořená písma. (Bývala tedy draze licencovaná, finanční omezení platilo nicméně hlavně pro druhou polovinu devadesátých let, dnes existuje i řada volně šiřitelných Type 1 písem). Stejně jako TrueType fonty jsou i Type 1 nepřenositelné mezi jednotlivými platformami. Přesto byla Type 1 písma po celá devadesátá léta a mnohde až dodnes používána jako nejlepší, ne-li jediná vhodná pro profesionální tisk, neboť šlo o písma sice drahá, ale precizně a důkladně vytvářená.

Type 1 fonty se také využívají v DTP systému TeX.

5.2.2.3 OpenType

Technologie OpenType je dalším vývojovým krokem ve vytváření široce uznávaných formátů písem, na kterém spolupracovaly dokonce i jinak konkurenční firmy Adobe a Microsoft. Formát OpenType má nahradit oba starší formáty a anulovat jejich nevýhody. Písma OpenType jsou tedy převoditelná mezi jednotlivými platformami (Windows, Mac OS, Linux...) a umožňují v rámci jediného fontu definovat až 65 536 znaků. OpenType písma tak nabízejí obrovské množství variant jednotlivých znaků (glyfů), velkou paletu znaků, jejich speciální formy (ligatury, různé varianty pro různé velikosti, iniciály aj.) i lepší práci s kerningem znaků. Bohužel pokročilá typografická nastavení a vlastnosti OpenType písem jsou zatím jen málo podporována jednotlivými textovými procesory.

OpenType fonty mají také další vlastnosti, které jim slibují velkou budoucnost: jsou dobře komprimovatelné, proto jednotlivé soubory nejsou příliš velké; jsou různě uzamykatelné, což vede k lepší ochraně autorských práv (umožňují uzamčení proti úpravám, ale třeba také proti exportu do PDF); umožňují tzv. subsetting, tj. možnost uložit v dokumentech jen využitě znaky, nikoliv jejich celou sadu aj.

OpenType písma fungují pouze na platformách schopných podporovat tzv. Unicode kódování, není proto možné je využít např. na starých Windows 95 nebo na starším Mac OS od Apple. (Výjimkou jsou tzv. OpenType TT fonty, které využívají zobrazovací metody starých TrueType písem – klasická OpenType písma jsou vykreslována vlastně postskriptově – a jsou proto použitelná i na starších Windows.)

5.2.3 Práce s písmem

Jednotlivé fonty jsou v počítači uživatele reprezentovány soubory s koncovkami, které jednoduše označují i jejich formát. Nejjednoznačnější jsou soubory OTF, značící OpenType písma. Koncovky TTF mohou kromě starších TrueType znamenat i tzv. OpenType TT fonty. Zatímco tyto dva formáty se většinou ukládají do předem daného systémového adresáře (složky; pod windows to bývá většinou složka C:\windows\fonts nebo odpovídající varianta), písma Type 1 formátu mívají vlastní

adresář a nechovají se přímo jako součást systému. Type 1 písma mívají koncovky PFB, resp. PFA, v závislosti na systému.

S těmito soubory lze pracovat jako s jakýmkoliv jinými soubory a jejich nalezení v systému počítače, jejich kopírování na jiné médium nebo odesílání emailem atd. je jedním ze základních požadavků pro ty uživatele, kteří musí svou práci sdílet s jinými. (Není snad třeba vysvětlovat, že až na výjimky má téměř každý uživatel ve svém počítači jinou množinu písem, nemluvě snad o takových „systémových“ písmech jako je Times New Roman či Arial.)

Pro správu, katalogizaci a organizaci fontů existuje celá řada programů, ať již placených, nebo šířených pod různými částečně nebo zcela bezplatnými licencemi (shareware, freeware, GNU). Aniž bychom zde chtěli vyzdvihnout nebo snad přímo doporučit nějaký program, je možné potenciální zájemce odkázat na klíčová slova jako „font viewer“, „font manager“, „font navigator“ atd.

Je nutné zdůraznit ještě jednu věc: máme-li písma ve starším formátu TTF, pak najdeme v počítači vlastní soubor pro každý řez písma a navíc i zvláštní soubory pro různá kódování písem, tj. např. pro středoevropské, východoevropské či pobaltské sady písem. Tyto soubory se pak vyznačují různými písmeny v názvu jako CE, EE, BAL atd. To vše je samozřejmě vynuceno omezením znakové sady na 256 znaků, jak bylo zmíněno výše.

Všimněme si také, že názvy jednotlivých souborů ne vždy korespondují s názvy jednotlivých fontů (viz obrázek 5.V). Naopak některé zkratky (bohužel však mezinárodně neustálené) v názvech jak fontů, tak jednotlivých souborů mohou naznačit uživateli, o jaké písmo, resp. jaký řez písma se jedná (sans vs. serif; Bd, Itc jako bold a italic; Dem pro demibold; Cond pro condensed, zúžené atd.).

 GillSans (TrueType)	PS_13872.TTF
 GillSans Bold (TrueType)	PS_13874.TTF
 GillSans Bold Italic (TrueType)	PS_13875.TTF
 GillSans Condensed (TrueType)	PS_14053.TTF
 GillSans Condensed Bold (TrueType)	PS_14054.TTF
 GillSans ExtraBold (TrueType)	PS_14051.TTF
 GillSans Italic (TrueType)	PS_13873.TTF
 GillSans Light (TrueType)	PS_13870.TTF
 GillSans Light Italic (TrueType)	PS_13871.TTF

5.V: (Ne)korelace mezi názvy fontů a jednotlivých souborů (zobrazení v Průzkumníku Windows)

5.2.4 Písmolijna

Firma, která se zabývá vytvářením počítačových písem, se obecně nazývá písmolijna (odkaz na historickou tradici odlévání olověných liter je zřejmý). Firem, které se u nás zabývají tvorbou či úpravami (počešťováním) písem, je několik. Mezi nejznámější patří Střešovická písmolijna/Storm Type Foundry, která „okupuje“ příznačnou adresu www.pismolijna.cz. Písmolijny obecně jsou schopné dodávat písma ve formátu, který si zákazník vyžádá, odklon od starších TrueType, respektive Type 1 písem ve prospěch dokonalejšího OpenType formátu je však zřetelný. Různé písmolijny mohou vytvářet své vlastní varianty známých písem, proto se např. i jednotlivé „Helvetica“ či „Timesy“ mohou lišit v závislosti na tvůrci.

6 TYPOGRAFIE A ZÁKLADNÍ TYPOGRAFICKÁ PRAVIDLA

V této kapitole shrnujeme pouze základní typografická pravidla a mechanismy tak, aby umožnily studentům bohemistiky, seznamujícím se se světem DTP, pustit se alespoň do jednoduché hladké a základní smíšené sazby, případně editorský a korektorský připravit k sazbě třeba dodaný rukopis. Pro zájemce o hlubší vhléd do problematiky typografie musíme odkázat na literaturu připojenou v kapitole 8, jmenovitě na výbornou a přehlednou publikaci *Praktická typografie*, kterou jsme i my nejčastěji konzultovali při vytváření tohoto přehledu. Základní pomůckou pro sazbu však mohou být i *Pravidla českého pravopisu*.

Správně zvládnutá typografie by měla pomoci k tomu, aby výsledek měl nejenom informační, ale také estetickou hodnotu. K tomu existuje celá řada uznávaných pravidel na úrovni buď jednotlivých znaků nebo větších celků, většinou odstavců. Tato pravidla sice nejsou nijak vymahatelná (tj. nakladateli, redakcemi atd., na rozdíl od starší doby, kdy existovala jediná norma), mělo by však být dobrým mravem všech, kdo pracují s počítačovou sazbou, se s nimi seznámit, a pokud není dobrého důvodu proti, také je dodržovat.

(Zastavme se ještě u názvosloví: obecně rozlišujeme sazbu hladkou, v níž je primární text a kterou je dále možné dělit na dílovou (knižní) a časopiseckou, sazbu smíšenou, ve které se využívá více způsobů vyznačování částí textů i grafických prvků, a sazbu akcidenční. Ta je typická pro tiskoviny a grafiky „jednorázové“, jako jsou plakáty, vizitky nebo podobné výstupy. Lze také odlišit sazbu pořadovou, tabulkovou, matematickou atd.)

6.1 Odstavcová sazba

U odstavcové sazby rozlišujeme především sazbu na (levý) praporek (lidově „doleva“) a sazbu do bloku s různě zarovnaným posledním, tzv. východovým řádkem. Odstavcová sazba na střed nebo na pravý praporek se využívá výjimečně, především u různých akcidenčních tisků (u knih se sazba na střed používá někdy u sazby poezie).

U odstavců nastavujeme a hlídáme tři základní prvky: velikost odstavcové zarážky (odsazení prvního řádku), velikost východového, posledního řádku a existenci tzv. parchantů (nebo též bastardů, u nás ovšem nejčastěji familiárně označovaných za tzv. vdovy a sirotky).

Odstavcovou zarážku počítáme v relativních hodnotách, nejčastěji v rozmezí 1 – 3 čtverčiků zvoleného písma (viz kapitola o písmu). Východový řádek by neměl být nikdy kratší, než je velikost odstavcové zarážky, což platí také pro mezeru mezi koncem východového řádku a pravým okrajem sazby (rámce sazby). Řešení obou problémů spočívá většinou v ručním upravení prostrkání znaků nebo dělení slov. Problém s „příliš dlouhým“ východovým řádkem lze také vyřešit vynuceným zarov-

náním celého odstavce do bloku, ideálně ovšem tak, aby výsledek nepůsobil rušivě (na konci textu nebo většího oddílu).

O tom, jak si poradit s parchanty, tj. „zapomenutými“ prvními (vdovy), respektive posledními (sirotci), řádky odstavců, si povíme něco v příští kapitole. Tamtéž jsou uvedena také pravidla pro dělení slov, která patří k odstavcové sazbě.

V rámci odstavců je třeba si dále hlídat tzv. řeky, což jsou příliš roztažené mezislovní mezery v několika po sobě následujících řádcích, takže vzniká optický negativní dojem bílého místa „protékajícího“ napříč řádky – „řeky“. Ta vzniká nejčastěji vypnutým dělením slov u odstavců nebo jeho špatnou aplikací tam, kde si počítačový algoritmus a slovník pro dělení nedokázaly poradit, což se může stát u cizích slov, toponym atd. Při druhém problému je nejčastějším řešením ruční úprava dělení. V závislosti na programu, ve kterém pracujeme, lze různým způsobem počítači sdělit, kde má dělit či naopak nedělit problematické slovo (postup v InDesignu je nastíněn opět v další kapitole).

6.1.1 Odstavcová vs. Jednořádková sazba (Adobe)

Tato poněkud záhadná terminologie se vztahuje k tomu, jak Adobe InDesign pracuje s typografií odstavce. Uživatel si totiž může zvolit, zda bude program při úpravě mezislovních mezer při zarovnávání textu do bloku zpracovávat a hodnotit jednotlivé řádky zvlášť, nebo zda bude brát v potaz celek, a tudíž jakoukoliv drobnou změnu v textu aplikovat na sazbu celého odstavce. Obecně lze říci, že odstavcová sazba dá v hladké sazbě lepší výsledky, uživatele však při prvním používání může zmást zdánlivě velký dopad drobných úprav (odstranění jediné mezery atd.) na sazbu celého odstavce, stejně jako někdy do jisté míry „tvrdohlavé“ chování programu při dělení slov.

6.2 Pravidla sazby jednotlivých znaků a prvků

řádky – Obecně platí, že jednotlivé řádky by neměly být zalomeny tak, aby končily tečkou, není-li tato tečka zároveň zakončením věty. Na konci řádku tedy nesmí stát zkratky, tituly, iniciály jména, řadové číslovky, části letopočtů (data), pomlčky aj. Zlom řádku by také neměl od sebe oddělovat číselné vyjádření různých hodnot a jejich značek (30 kg, 20 % atd.), peněžní hodnoty, trojčíslí telefonních čísel nebo čísel vyšších řádů (desetitisíce a výše; čtyřciferná jen někdy).

neslabičné předložky a jednoznakové spojky – S výjimkou malého „a“ platí, že se nesmí objevovat na koncích řádků; někteří typografové odsouvají na další řádek všechny znaky bez výjimky, tedy i malé „a“. Trochu jiná pravidla platí pro velmi úzkou sazbu (do 25 znaků), kde jsou slabičné spojky většinou tolerovány.

datum a letopočet – Den je vždy vyjádřen arabskou číslicí s tečkou, měsíc buď slovně, nebo arabskou či římskou číslicí s tečkou, letopočet se sází celý, tj. bez vynechávání prvního dvojčíslí. Datum, jak již bylo řečeno, se nesmí rozdělovat mezi dva řádky.

čas – Hodiny a minuty jsou od sebe odděleny tečkou bez mezer (12.00), minuty a sekundy pak dvojtečkou (12.05:03). Desetiny sekund se od celých sekund oddělují čárkou (19:26,3 min.). Slova „hodina, minuta, sekunda“ se krátí značkami h, min, s nebo zkratkami h., hod., min.

Ačkoliv Jazyková poradna Ústavu pro jazyk český AV ČR udává, že podle ČSN 01 6910 z roku 2007 se časové údaje oddělují dvojtečkou, za níž se nedělá mezera (03:25, 10:30:15, dostavte se v 7:15 h.), a lze tedy připustit psaní s tečkou i dvojtečkou, doporučujeme užívat první, starší a osvědčený typografický úzus. Ten mimo jiné umožňuje se rychleji zorientovat v tom, které časové hodnoty ve vysázeném textu máme. Oddělování dvojtečkou je naopak běžné u sportovních výsledků (doběhl v čase 2:03:15,3 – bez označení časových jednotek).

závorky – Dávají se do nich ty části textu, které jsou do věty volně vloženy a nejsou její přímou součástí. V hladké sazbě se používají závorky oblé (), hranaté [] a výjimečně složené (svorky) { }. Dává-li se do závorek celá věta, sází se tečka uvnitř závorek, pokud je v závorce pouze část věty, sází se tečka až za závorku.

Hranaté závorky se uzualně používají pro zaznamenávání výslovnosti, při vypouštění části textu v citacích nebo při vkládání komentářů k citacím. V některých specializovaných, např. právnických textech existuje nicméně úzus používat hranaté závorky v souladu s matematickými pravidly, tj. hranaté závorky jsou nadřazené. V případě závorek v závorkách jsou tedy závorky vnitřní kulaté, závorky vnější hranaté.

Nepoužíváme vytváření závorek pomocí lomítka („/“), a to i přesto, že se tato varianta objevuje v *Pravidlech*.

značky ve spojení s číselnými hodnotami – Obecně platí, že značky se od čísla oddělují nezúženou mezerou: 10 ha (10 hektarů; pozor na rozdíl mezi substantivním a adjektivním výrazem: 10ha se čte jako desetihektarový), 3 kg, 14 % (čtrnáct procent vs. 14% = čtrnáctiprocentní), § 9 = paragraf 9 (v případě více paragrafů se značka nezdvouje), 100 kWh = 100 kilowatthodin, rychlost 50 km/h.

NB: Co se vypisování adjektiv týče (10ha, 14%), pak slovně tyto výrazy píšeme bez mezer, bez spojovníku a nepisujeme do nich koncové části číslovek, neboť slabiky -ti a -mi- jsou už obsaženy v tvaru číslovky. Koncové -ti a -mi nepíšeme ani u základních číslovek: pro děti od 14 let, za účasti 58 % voličů, od 5 do 8 h (nikoli „od 5ti do 8mi“ nebo „od 5-ti do 8-mi“).

peněžní částky, měna – Číslo a značku měny oddělujeme mezerou, značka může stát před nebo za číselným údajem. Trojice čísel oddělujeme pevnou (neoddělitelnou) mezerou, tak aby se číslo nezalomilo na více řádků. Nulu je možné nahradit pomlčkou (20,-), neměla by pak ale následovat značka měny. Způsob zápisu 1 000 000,- Kč je tedy chybný. Správně píšeme 1 000 000 Kč nebo Kč 1 000 000,- nebo i Kč 1 000 000. Uvádíme-li haléře, oddělujeme jejich hodnotu čárkou: 1 000 000,00 Kč.

ÚJČ AV ČR povoluje také oddělování jednotlivých trojčíslí finančních částek tečkou („z bezpečnostních důvodů“), v praxi se však s takovým zápisem setkáváme spíše u specializovaných ekonomických či účetních záznamů a rozhodně jej nedoporučujeme v sazbě např. publicistiky nebo beletrie.

stupně a teplota – Vyjadřuje se pomocí znaku pro stupně ($^{\circ}$), který se sází s mezerou za číslem, je-li ve spojení se znakem „C“ pro „Celsia“, jinak bez mezer. Je-li součástí výrazu znaménko hodnoty, sází se bez mezery před číslo: $+24^{\circ}\text{C}$, -1°C . Úhly označujeme kombinací stupně, minuty, vteřiny ve formátu $22^{\circ}05'11''$ – tj. bez mezer.

matematické výrazy – Sázíme s nezúženou mezerou mezi jednotlivými symboly: $2 + 5 = 7$ atd. Pozor na existenci znaku pro „krát“, tj. „ $\times$ “, který nebudeme zaměňovat s literou „x“, případně s hvězdičkou „*“ (s možnou výjimkou pro počítače, respektive programy, které využívají klávesu „*“ jako matematickou funkci „krát“). Pokud je $\times$ součástí jednoho složeného slova společně s číslem, píšeme tyto výrazy dohromady: $20\times = 20\text{krát}$, tj. dvacetkrát. Vyjadřují-li znaménka $+$, $-$ kladnou nebo zápornou hodnotu čísla, přiléhají k číslici bez mezer, stejně jako u stupňů. Znak $\times$ používáme také pro udávání rozměrů, a to s mezerami ($20 \times 15\text{ cm}$). Pro dělení užíváme „:“ nebo „/“, nezapomínáme na samostatný znak pro plusminus: „ $\pm$ “. Nemá-li písmo definovaný samostatný glyf pro minus, používáme pomlčku, měla by však mít posunutý účař, tak aby bylo do roviny s horizontální příčkou znaménka $+$.

uvozovky – V češtině užíváme tři varianty uvozovek: dvojité „ “ “, jednoduché „ ' “ a případně boční „ « “. Jako základní však slouží pouze uvozovky typu 9966 (tj. dvojité „ “ “), je proto třeba dávat pozor na různé špatné varianty („ ” “ apod.). Druhé dva typy uvozovek se užívají pouze v případech, kdy do uvozeného textu potřebujeme vložit ještě další uvozovky (např. v případě přímé řeči uvnitř přímé řeči). Tzv. boční uvozovky však používáme spíše výjimečně. Pozor na znak pro „palce“, tj. měrnou jednotku, který připomíná uvozovky, nikoliv však typografické: $17''$ vs. $17''$ (sedmnáctipalcový).

výpustka – Jinak známá jako trojtečka, lidově „tři tečky“. V typografii se jedná o jediný znak, který má mnoho funkcí, nejčastěji je používán v místech, kde je vynechána určitá část textu, případně znázorňuje nedořečenost výpovědi či přerývanost řeči. Sazba je mírně komplikovaná: nikdy se nespojuje s další tečkou, otázka mezer okolo je však sémantická. Tam, kde je naznačeno vynechání textu, je výpustka oddělena mezerou, avšak tam, kde naznačuje nedořečenost, se připojuje přímo. Při použití trojtečky ve výčtech může za trojtečkou následovat přímo čárka. V citacích, jde-li o naznačení vypuštění nějakého úseku, se výpustka většinou uzavírá do hranatých závorek.

pomlčka – Je často zaměňována za spojovník, ačkoliv je delší, zabírá polovinu čtverčíku (dlouhá pomlčka je pak na délku celého čtverčíku). Nejčastěji se používá pro označení rozsahu, méně často pro jiné funkce, jako je uvozování

přímé řeči, používání v zápisu finančních částek (20,-) atd. Pomlčku v textu oddělujeme z obou stran mezerami, pokud funguje jako interpunkční znaménko (čárka, závorka). Jsou-li oba oddělené výrazy jednoslovné, především ve významu od-do, píšeme pomlčku bez mezer (např. otevírací doba 8-16, s. 6-18, 1929-1960). Pokud je alespoň jeden z výrazů víceslovný, pomlčku mezerami oddělujeme (např. autobus Hradec Králové - Litomyšl, stejně tak rozepsané datum 1. červen - 15. srpen). Stojí-li pomlčka na konci řádku, na dalším se (na rozdíl od spojovníku) neopakuje - pomlčka až na výjimky, například když zastupuje uvozovky, nesmí být na začátku řádku. Použijeme-li pomlčky ve větě pro oddělení vsuvky atd., může po druhé pomlčce následovat čárka, jinak je ale možné čárku vynechat, nenaruší-li se tak příliš zřetelnost výstavby věty.

dlouhá pomlčka - Glyf, který se používá málokdy, je běžnější pro anglickou sazbu (—). Přesto ji lze nalézt i u nás, buď jako zástupný znak pro uvozovky při zahájení přímé řeči, nebo při sazbě např. poezie (naznačující často zámlku, odmlku).

spojovník - Typograficky též „divis“. Od pomlčky je odlišen jak graficky („-“; tj. jen třetina čtverčíku), tak především funkčně. Spojovník se používá pro vyjádření těsného vztahu mezi různými substantivy nebo adjektivy (Rimskij-Korsakov, černo-bílý). Nejčastěji však pro připojení spojky -li, většinou k prvnímu slov(es)u věty: je-li, mohl-li. Pro přesná pravidla používání spojovníku doporučujeme nastudovat příslušnou pasáž *Pravidel*, případně jejich aktualizace. Ke specifickým typografickým problémům lze uvést, že spojovník se používá u výrazů, jejich součástí jsou nedokončená slova (dvou- až třílůžkový pokoj, resp. 2- až 3lůžkový pokoj); při snaze zdůraznit význam části slova (ne-bytí) nebo segmentaci slov (sa-mo-ob-slu-ha); při sazbě specifických lingvistických textů (slova zakončená na -ec nebo -tel).

značky †, * - Symboly pro úmrtí, resp. narození, oddělujeme od letopočtu nebo datumu zúženou mezerou.

lomítko - Pokud lomítko („/") slouží jako oddělovací znaménko při výčtu jednotlivých výrazů, odděluje se mezerami. Při jiných významech (km/h, akademický rok 2007/2008) se sází bez mezer. Některé příručky trvají na tom, že lomítko se opakuje na začátku řádku, končí-li předešlý řádek lomítkem, mnozí však toto pravidlo opouští ve prospěch jediného lomítka na konci řádku.

Při zjednodušené sazbě (přepisu) poezie nebo písňových textů atd. se používá jedno lomítko jako oddělovač veršů, dvě lomítka jako oddělovače strof. Lomítko se také používá pro zápis zlomků, většina počítačových písem však umožňuje také zápis zlomků pomocí specializovaných glyfů-slitků: ½, ¼, ¾ aj. (počet těchto glyfů je však většinou velmi malý).

tečka na konci věty - Platí, že věta nesmí končit dvěma tečkami. Je-li věta tedy ukončena například zkratkou nebo akademickým titulem, další tečku po ní nepíšeme.

Na závěr musíme dodat jednu poznámku. Jak bylo řečeno v předchozí kapitole, v typografii a sazbě rozeznáváme několik velikostí mezer. Mezery vlasové a osminkové slouží především a hlavně k tomu, aby výsledný produkt nabízel vyvážený estetický požitek. Je nutné se tedy občas uchýlit k jejich použití, a to i tam, kde „pravopisné“ pravidlo říká, že by mezera vložena být neměla. Například v místech, kde hrozí slitky znaků nebo tam, kde je vlivem vyznačování úseků textu sazba nepřehledná. Velmi často se stává, že znaky vysázené kurzívou rušivě zasahují do okolí (kurzívní znaky v závorce mají tendenci vytvářet se závorkou slitky atd.), že pomlčky se vizuálně přimykají více k některým znakům (viz rozsah 1–0), že dochází ke ztrátě distinktivnosti znaků v horním indexu atd. V těchto případech přichází na řadu sazeč-typograf a jeho úsudek.

Je nutné také dávat pozor na vyznačování částí textů – jsou-li například termíny v závorkách vyznačovány kurzívou, závorka sama bude většinou vysázena základním řezem, stejně jako třeba čárky oddělující výčet položek zdůrazněných tučným řezem apod.

7 AUTOMATIZOVANÉ POSTUPY PŘI SAZBĚ

7.1 Styly

Práce se styly je jedním z postupů, které při správné aplikaci dokáží nejen ušetřit čas, ale také zjednodušit a zautomatizovat práci s textem a sazbou. Styly mohou být také nápomocny při kontrole některých prvků sazby, jako je umisťování nadpisů kapitoly na začátky stránek (jako je tomu zde), zabezpečení toku textu (ochrana před bastardy) či kontrola návaznosti hierarchie textu, jak bude ukázáno níže. Styly jsou také základním prvkem pro vytváření obsahů a mohou být klíčem pro využívání některých vestavěných funkcí DTP programů, jako je funkce Hledat/Nahradit.

7.1.2 Definice stylu

Styl je možno definovat jako skupinu atributů (charakteristik), které jsou platné pro určitý úsek textu, resp. pro tabulky nebo jejich části či pro grafické prvky.

7.1.3 Druhy stylů

Pro práci s textem jsou základní dva druhy stylů: styl odstavcový (paragraph style) a styl znakový (symbol style). Existuje také styl grafických objektů (např. styl obdélníků, čar, šipek atd.), který je primární při práci s grafickými programy, ale který se objevuje také v novějších verzích DTP programů. Některé programy (Word, InDesign od poslední verze) pracují také se styly tabulek, kde se definované atributy aplikují na buňky, řádky atd.

Jak již z jejich názvu vyplývá, první z nich ovlivňuje vzhled celého odstavce, druhý je aplikovatelný na jednotlivé znaky nebo skupiny znaků, tj. jednotlivá písmena, slova či skupiny slov.

Pro názornost vezmeme za příklad opět toto skriptum: mezi odstavcové styly, které jsou zde využívány, patří:

Základní text; tímto stylem je psána většina textu, například i tato věta.

**NADPIS KAPITOLY: STYL VIDITELNÝ
V ÚVODU STRÁNKY, TJ. »AUTOMATIZOVANÉ
POSTUPY...«**

Nadpis 2: styl aplikovaný na podnadpisy »Definice stylu« a »Druhy stylů«; jak jeho název napovídá, je zařaditelný do určité hierarchie stylů (Nadpis 1, Nadpis 2, Nadpis 3 atd.).

Specifické znakové styly nejsou na této stránce využity, i běžný uživatel je však zná například jako styl „Znak poznámky pod čarou“, kde je v rámci daného stylu definováno, jak má vypadat značka odkazu umístěná do textu, který je jinak definován svým vlastním stylem. Znakové styly se jinak nejčastěji využívají, pokud je do „normálního“ textu nutné vložit například slova psaná jinými abecedami – řeckou, hebrejskou, azbukou; či pokud využíváme v rámci jednoho odstavce pro některá slova odlišná písmena (fonty). Velmi často se definuje znakový styl pro polotučné

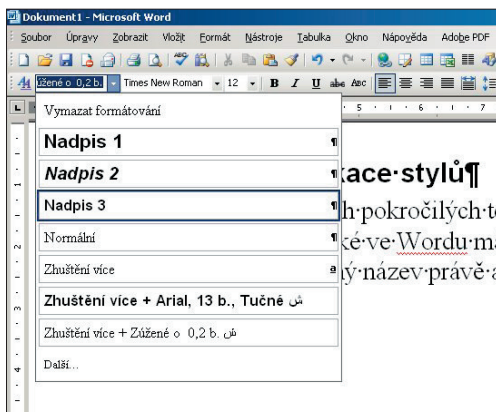
písmo ve chvílích, kdy dané písmo takový řez obsahuje. Občas je nutné definovat znakový styl pro specifické znaky, které některá písma nemají, a při sazbě je pak nutné si je „vypůjčit“ z jiného fontu.

Ještě snad jeden názorný příklad: pokud připravujeme publikaci například v písmu Baskerville OldFace DEE, pak zjistíme, že neobsahuje symbol eura, tj. €. (Při importu z Wordu nebo jiného textového procesoru se tak v DTP programu objeví zástupný znak, značící chybu, v InDesignu tedy „čtvereček“.) Nyní můžeme: a) daný znak vybrat a font Baskerville OldFace DEE ručně vyměnit za příbuzný Baskerville OldFace OTF, který již symbol eura obsahuje; b) tento postup dokončit tím, že již nahrazený, správný znak označíme jako nový znakový styl, který si uložíme. Tím získáme jednu zásadní výhodu – při další potřebě nahrazení jednoho fontu za druhý můžeme vše urychlit pouhým jedním kliknutím na nově vytvořený znakový styl. Tento postup je nejvýhodnější, pokud je sazba prováděna v méně standardním fontu, ovšem originál je vytvořen v textovém procesoru, který si sám automaticky dokáže s problematickými znaky poradit, jak je tomu například u Wordu. InDesign a jiné sazecí programy nejsou k záměnám fontů takto benevolentní, proto se mnohdy stává, že text, který se jeví jako dokonalý ve Wordu apod., je po importu do InDesignu plný „čtverečků“ – chybějících či vadných glyfů.

Stejný problém nastane, pokud aplikujeme jeden odstavcový styl na text, který je z podstaty heterogenní, tj. který obsahuje znaky z více fontů, byť tento problém mnohdy řeší OpenType fonty s jejich širokou paletou znaků a možností.

7.1.3 Vytváření a aplikace stylů

Styly jsou dnes součástí všech pokročilých textových procesorů a všech DTP programů. Ačkoliv si to málokdo uvědomuje, také ve Wordu má každý odstavec a/nebo znak přiřazen určitý styl. Okénko voleb stylů a samotný název právě aktuálního stylu je hned prvními položkami na paletě „Formát“ (viz obr. 7.1); při klepnutí na první ikonku je také možné vyvolat postranní paletku Styly a formátování (od verze 2003). Pokud rozbalíme (tj. klikneme na „zobáček“) roletku stylů, Word nám nabídne dostupné, přednastavené odstavcové styly. Pokud ji rozbalíme se současným přidržením klávesy Shift, Word nám ukáže celou škálu přednastavených stylů, které jsou v danou chvíli dostupné. (V boční roletce je možné filtrovat styly dle více parametrů.) Všimněme si, že Word naznačuje, do jaké skupiny



7.1: Roletka Styly ve Wordu

jednotlivé styly patří pomocí různých symbolů vpravo od názvu stylu; jejich smysl je jednoduché dešifrovat.

Chceme-li si vytvořit nový vlastní styl, existuje několik možností, z nichž nejjednodušší jsou tyto:

1. využití ikony „Nový styl“ na panelu „Styly a formátování“ (u starších verzí, případně není-li panel dostupný, přes menu Formát a položku Styly/Styly a formátování), kdy se uživatel dostane do dialogu, ve kterém postupně ručně nastaví všechny důležité atributy zamýšleného stylu;
2. opačný postup – nejdříve ručně vybereme odstavec nebo alespoň část textu a ručně nastavíme všechny zamýšlené vlastnosti (přes formátovací ikony, klávesové zkratky atd.) a poté klikneme na ikonu „Nový styl“ – vytvoří se nový styl, který již bude obsahovat všechny ručně definované atributy.

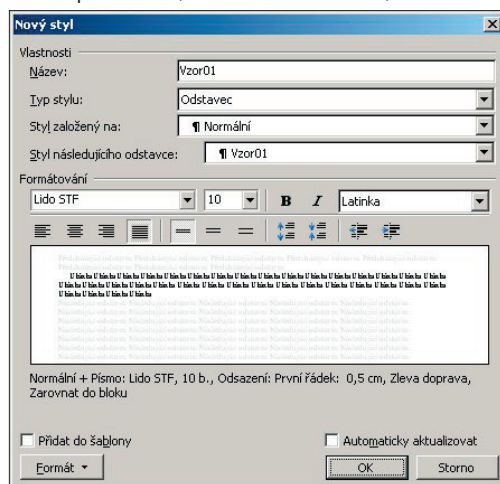
Stejný postup je obecně platný u většiny textových procesorů a DTP programů s jistými grafickými a terminologickými rozdíly. V InDesignu pracujeme s roletkou „Odstavcové styly“ (je sdílená se Znakovými styly a Vzorníkem), ve které máme samostatnou drobnou ikonu pro „Nový styl“, případně se můžeme dostat k více volbám přes „zobáček“ rozšiřující nabídky.

Základní dialogové okno pro vytváření vlastních stylů (zde nazvaný „vzor 01“) v obou programech máme na obrázcích 7.II a 7.III.

Vidíme, že v obou oknech máme pro orientaci textový popis stylu, ze kterého je možné vyčíst nejdůležitější charakteristiku (typ písma, velikost, zarovnání a odsazení odstavce, případnou dědičnost atd.), další jednotlivé volby jsou různé skryty a my si z nich ukážeme či vysvětlíme jen ty nejdůležitější (na příkladu InDesignu).

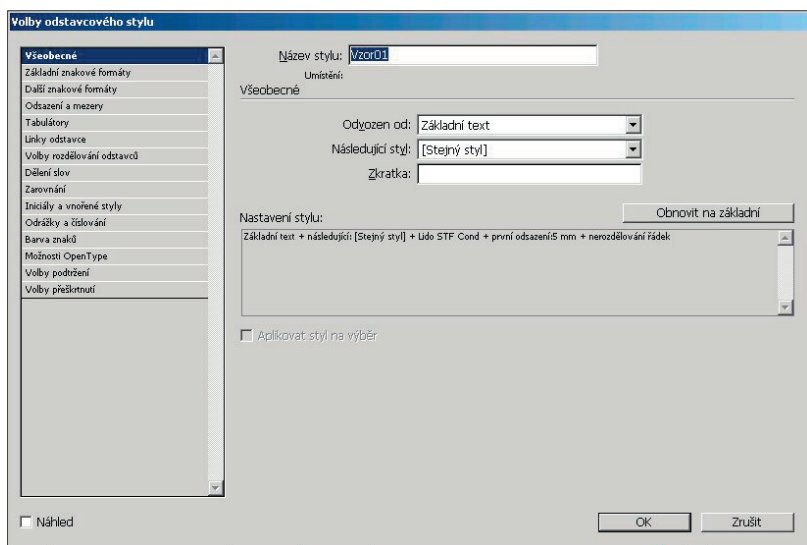
7.1.4 Vybrané položky z definice stylu

Na první kartě, nazvané všeobecné, se kromě názvu stylu definuje především to,



Obr. 7.II: Vytváření odstavcového stylu ve Wordu

zda je nově vytvářený styl odvozen od jiného. Tato vlastnost s sebou přináší poměrně rafinované možnosti a důsledky, které vycházejí z tzv. dědičnosti stylů. Pokud totiž založíme nový styl na základě jiného, již definovaného, pak náš nový styl přebere všechny vlastnosti původního stylu až na ty, které specificky zadáme jako odlišné. Pokud bychom tedy vytvořili styl „Vzor 02“ a odvodili jej od již definovaného stylu Vzor 01, máme dva totožné styly; pokud v dalším



7.III: Vytváření odstavcového stylu v InDesignu

roku pouze nastavíme, že náš nový styl Vzor 02 bude mít velikost písma 14 bodů, pak je tato vlastnost jedinou, ve které se odlišují – font, řez, zarovnání a odsazení se v novém stylu zkopírují ze stylu, ze kterého odvozujeme. Kromě toho ovšem, a to je velmi důležité, změníme-li později například zarovnání ve stylu „Vzor 01“, od něj odvozený „Vzor 02“ tuto změnu též převeze a ponechá si jako jedinou odlišnou vlastnost jinou velikost písma! Toto odvozování vlastností jednotlivých stylů lze využít v náš prospěch, může však i přinést problémy, pokud jsme si definování více stylů zjednodušili vícenávaznou dědičností.

Položka „Následující styl“ potom určuje, v jakém stylu se bude sázet odstavec, který bude následovat po ukončení aktivního odstavce, tj. po stylu „Nadpis“ bude pravděpodobně následovat styl „Podnadpis“ nebo „Normální text“ atd. Toto nava-zování je ovšem platné jen tehdy, pokud text přímo píšeme, tj. program bude sledovat, kdy ukončíme odstavec stisknutím Enter a další odstavec bude automaticky formátovat podle vybraného stylu. Při zpětném formátování textu, který jsme do InDesignu vložili ze schránky, nebo při importu textu tato návaznost nefunguje – neplatí tedy, že když označíme jeden odstavec kliknutím jako „Nadpis“, že by se následující automaticky zformátoval jako „Normální text“ atd.

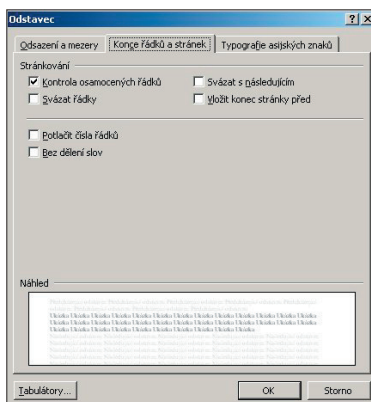
Další záložky, tj. formátování znaků a nastavování odsazení celého odstavce, případně jeho zarovnání, by žádné problémy interpretací a nastavením přinášet neměly, stejně jako definování tabulátorů či linek kolem odstavců.

Zajímavé volby jsou ovšem dostupné pod položkou „Volby oddělování odstavců“. Nastavíme-li volbu, že daný styl se musí spojit s následujícími řádky (nej-

častější volba je 2), pak zajistíme, že některé odstavce nebudou od sebe odtrženy, vychází-li mezera mezi nimi například na konec stránky. V praxi je to metoda, jak zajistit, že libovolný nadpis nebude odtržen od textu, který uvozuj; že se na konci stránky nevyskytne třeba název článku, zatímco jméno autora bude až na straně následující atd.

Volbou „Nerozdělovat řádky“ si zajistíme, že se v textu nebudou vyskytovat již jednou zmiňované „vdovy a sirotci“, tj. samostatné řádky na začátku, resp. konci odstavce, v místech, kde se láme strana. (Ve Wordu je tato volba skryta pod možná jednodušší název „Kontrola osamocených řádků“, viz obr. 7.IV.) Tento typografický nešvar je snad možné tolerovat u zúžené časopisecké či novinové sazby ve sloupcích, nikoliv však v klasické knižní sazbě. Volba „Nerozdělovat všechny řádky odstavce“ je méně častá, může se použít u citací, veršových slok atd.

Konečně nastavení „Začátku odstavce“ nám může zajistit, že specifické texty budou mít svoje fixní místo v sazbě, tj. například název kapitol bude vždy začínat na další straně atd. Všechny tyto relativně jednoduché volby uživatel ocení, pracuje-li s dlouhými texty, složenými z mnoha kapitol a podkapitol, ve kterých dochází k průběžným úpravám po celé délce textu.



7.IV: Kontrola „osamocených řádků“ ve Wordu“

7.2 Dělení slov

Jednou z položek při definování odstavcových stylů a zároveň funkcí, ke které se často dostaneme i při jednoduché sazbě bez využití stylů, je definování možnosti a limitů dělení slov. V češtině se obecně doporučuje nedělit slova o méně než pět písmenec a dále nedělit slova tak, aby na koncích nebo začátcích řádků zůstávala jen dvě písmena (což je mimochodem implicitní nastavení InDesignu). Je tedy vhodné nastavit dělení na distribuci „2 a 3“ znaky, tj. umožnit dělení po prvních dvou písmenech potenciálního děleného slova, resp. před jeho posledními třemi. Je ale třeba počítat s tím, že InDesign chápe interpunkční znaky za slovy, včetně tečky, otazníku nebo vykřičníku na koncích vět jako součást posledního slova věty!

K dalším pravidlům kolem dělení slov v češtině patří, že by se pod sebou neměla dělit více než tři slova (volba „Limit počtu dělení“) a samozřejmě by se slova neměla dělit nevhodně. Tím je míněno, že na koncích nebo začátcích řádků nesmí zůstat matoucí či vulgární slova (knihovna, nádržka atd.). Tento problém obecně žádný algoritmus nezkontroluje, snad s výjimkou vestavěných slovníků jednotlivých programů, kde jsou některá dělení notoricky známých slov zakázána.

Automatické dělení slov samozřejmě nemusí vždy fungovat bezchybně. Pokud si počítač nebude s některými slovy vědět rady, ať již je nerozdělí vůbec, nebo naopak špatně, je nutné do procesu vstoupit ručními úpravami. Nejjednodušší je počítači naznačit, kde se má dané slovo dělit – na správnou hranici dvou slabik vložíme tzv. neviditelný rozdělovník. Ten najdeme v InDesignu přes menu Text, položku Vložit speciální znak (nebo samozřejmě přes příslušnou klávesovou zkratku, implicitně Shift + CTRL + „-“ (pomlčka). Pokud se znak neviditelného rozdělovníku vloží před dané slovo, zakážeme tím jeho dělení!

7.3 Vzorové stránky

Jedním z důležitých prvků automatizace a zjednodušení sazby, především u komplexně strukturovaných knih, jako jsou odborné příručky, sborníky, manuály atd., je využívání tzv. vzorových stran (master pages).

Zatímco uživatelé pracující s textovými procesory, byť pokročilými, nejsou většinou zvyklí využívat vzorové stránky, tj. jakési šablony pro sazbu, v DTP programech může jít o poměrně důležitý prvek. Součástí definice vzorových stránek bývá nejčastěji nastavení záhlaví a zápatí (včetně např. číslování stran), vodících linek, okrajů sazby (tj. nastavení zrcadla sazby) či případných dekorativních a doprovodných prvků, které se mají promítnout buď v celé knize, nebo v některých jejích částech. Jako příklad si můžeme vzít tuto knihu: ačkoliv neobsahuje mnoho grafických prvků, přesto se evidentně liší liché (pravé) a sudé strany použitím navigačního prvku se jménem kapitoly. Prvních pět stran je navíc oproštěno jak od tohoto políčka, tak od zápatí ve formě číslování stran. Máme zde tedy vzorové stránky A (liché strany), B (sudé strany) a C (prvních pět stran).

Zatímco starší a jednodušší DTP programy jako PageMaker umožňovaly vytvořit pouze dvě vzorové stránky, jednu pro pravou a jednu pro levou stranu sazby, modernější programy nabízejí mnohem větší možnosti a variabilitu.

Aplikace vzorových stran na běžné stránky probíhá buď při vytváření nových stran, nebo i zpětně přes rozšířenou volbu „Aplikovat vzor na stránky“. Ne všechny strany v InDesignu však musí mít svou vzorovou stranu (stejně jako ne každý text musí mít nějaký určitý odstavcový nebo znakový styl)!

Dodejme ještě pár detailů. Číslo stránky se u InDesignu vkládá pomocí zkratky Shift + CTRL + N, resp. přes menu Formát, položku Vložit speciální znak ▢ Vložit číslo stránky. Volby číslování stran lze nastavit přes Formát ▢ Volby číslování a oddílů.

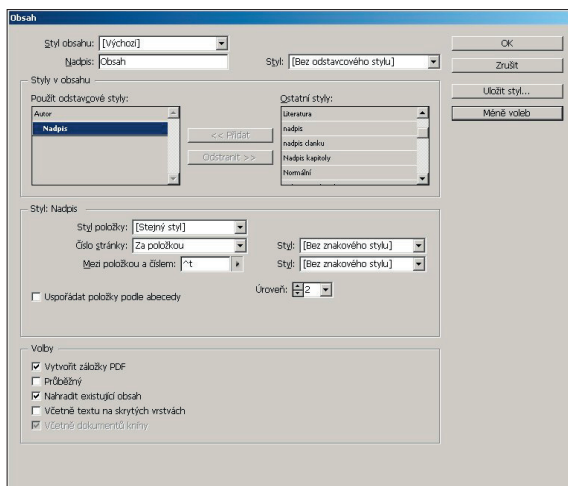
Určité možnosti vytvářet jakési „pseudo vzorové stránky“ umožňuje i např. Microsoft Word, celý dokument by však bylo u něj nutné rozdělit na tzv. oddíly (viz menu Soubor, položka Vzhled stránky a záložka Rozložení; resp. menu Vložit, položka Konec a možnosti Typy konců oddílů). Tím je možné nastavit jednotlivým oddílům a/nebo sudým a lichým stranám jiná záhlaví, orientaci dokumentu atd. Svou primitivností a jistou těžkopádností nastavení však tyto možnosti nemohou soupeřit s použitím vzorových stránek, jak je chápou DTP programy.

7.4 Obsahy a rejstříky

Generování obsahu knihy, ať již jde o jeden dokument nebo více dokumentů spojených v jednu knihu, je v sázecích programech zautomatizováno, pokud jste při sazbě použili a správně aplikovali odstavcové styly. Předpokládejme, že máme třistastránkový dokument – sborník studií, který zahrnuje 20 patnáctistránkových studií. Každá z nich má na začátku uvedený svůj název a svého autora a ti jsou naformátováni za použití odstavcových stylů nazvaných např. „Nadpis“ a „Autor“. Potom je automatické vytvoření jednoduché: z menu Formát vybereme položku Obsah... a v nabízeném dialogovém okně si vybereme, které odstavcové styly se mají zahrnout do vytvářeného obsahu, jaká je mezi nimi hierarchie, zda a jak vložit číslo stránky, na které se vyskytuje námi označený odstavcový styl (resp. text podle něj naformátovaný), jaký vodící znak (tečka, podtržítiko atd.) se má případně vyskytnout mezi posledním znakem položky obsahu a číslem stránky jejího výskytu atd.

Ačkoliv slovní popis vypadá komplikovaně, ve skutečnosti je nastavení poměrně intuitivní. Níže přiložený obrázek, odrážející reálnou sazbu knihy, snad dokáže napovědět.

Při generování obsahu je důležité si uvědomit jednu věc – pravděpodobně nebudeme chtít, aby se nám všechny položky naformátované styly „autor“ a „nadpis“ v obsahu ukázaly právě v těchto



7.V: Vytváření obsahu pomocí stylů

formátech (předpokládejme, že půjde o větší velikost písma, často tučný řez, odstavec zarovnaný na střed sazby a s odsazením před nebo za atd.). Při vytváření obsahů je tedy možné a někdy důležité jednotlivé prvky přeformátovat (viz volba Styl položky). Styly a definice obsahu lze také ukládat nezávisle na dokumentu (viz menu Formát → Styl obsahu).

Na rozdíl od generování obsahů je tvoření rejstříků přece jen komplikovanější záležitostí, neboť je podmíněno tím, že autor nebo sazeč si během průběhu práce sám vybere a označí (otaguje) slova, která se mají následně v rejstříku zobrazit. Navíc ve flexibilitě a možnosti využití těchto automatizujících prvků sázecí programy dlouhou dobu pokulhávaly za Microsoft Wordem, jehož široké využití tzv. polí (menu Vložit, položka Pole a následující volby) umožňovaly již od verze 97 využívat celou

řadu jednoduše dostupných automatických funkcí. Přesto jsou i např. v InDesignu od verze CS (3.0) implementovány určité možnosti vytvářet indexovaný text a následně automaticky generovat rejstříky. K volbě nastavení a vytváření rejstříků, o níž se zde více rozepisovat nebudeme, se dostaneme přes vlastní okno Rejstřík (menu Okno, položka Text a tabulky, podpoložka Rejstřík).

7.5 Vyhledávání a nahrazování

Mezi postupy, které mohou pomoci s úpravami textu, patří také notoricky známé funkce Vyhledat, respektive Nahradit. O jejich existenci ví asi každý uživatel, málokdo však využívá všech možností, které v některých programech tyto funkce nabízejí, především při úpravě textu před vlastní sazbou, případně pro hromadné změny při sazbě nebo zlomových korekturách.

Spousta podkladů, které DTP pracovníci dostávají do rukou, odráží dlouhodobé (a většinou špatné) typografické návyky svých autorů: jednotlivé odstavce bývají „odsazeny“ několika mezerami nebo tabulátorem nebo dokonce dvěma; odstavce bývají oddělovány volným řádkem (tj. prázdným odstavcem, „Enter“); místo trojtečky (výpustky) jsou v textu tři jednotlivé tečky; pomlčky jsou zaměněny za spojovníky atd. apod. Dalším častým autorským nešvarem je vyznačování částí textů podtrháváním, jedním z mnoha přežitků psaní na psacím stroji. Mnozí s oblibou vytváří „tabulky“ v textu pomocí libovolného počtu tabulátorů, tak aby jim vyšel optický dojem zarovnaných sloupců atd. (Kromě toho samozřejmě existuje spousta prohřešků, které nám žádná automatická funkce opravit nepomůže, namátkou třeba ruční „prostrkávání“ písem pomocí mezerníku).

Jak lze tedy funkci Vyhledat/Nahradit využít k odstranění těchto problémů? Většina moderních programů, se kterými se v rámci tohoto skriptu i v rámci oboru setkáváme, umí vyhledávat nejenom jednotlivé litery, ale také tzv. skryté znaky, metaznaky, což jsou vlastně symboly zastupující klávesy a funkce: tabulátor, konec odstavce („Enter“) nebo spousta jiných.

Námi vzpomenutý příklad odsazování prvních řádků odstavců, tj. vytváření odstavcové zarážky pomocí tabulátoru, lze pak do sekvence znaků a metaznaků přepsat asi takto:

*poslední znak předchozího odstavce * znak konce odstavce * tabulátor * první znak nového odstavce*

V grafickém znázornění by pak tento text vypadal takto:

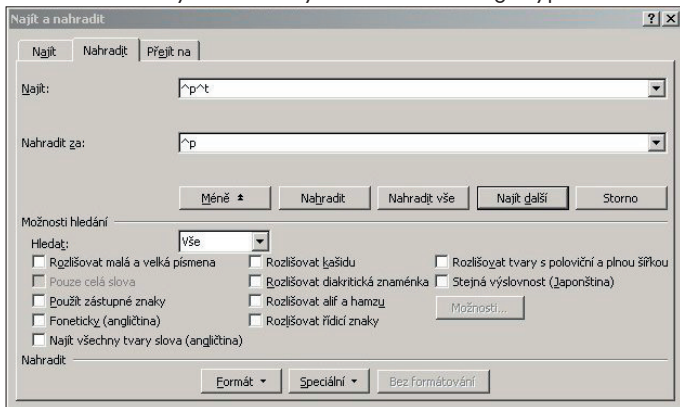
barevnou hloubku, resp. že ne všechny di;
zobrazující jen 4 či 16 barev. ¶
→ Nevýhodou při udávání barevných
číslech (zmiňované 4 či 16 či 256, což je

(Jde o způsob zobrazení skrytých znaků v MS Wordu. Tento režim zapneme/vypneme kliknutím na tlačítko znázorňující symbol konce odstavce na liště nástrojů „Standardní“ – ikona ¶.)

Jsmeli schopni jistého malého stupně abstrakce, pak je jasné, že tuto sekvenci znaků musíme nahradit sekvencí:

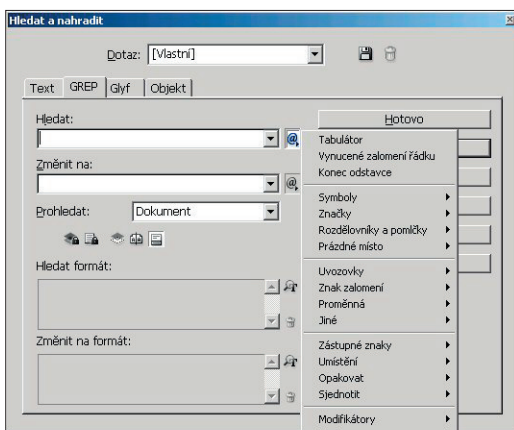
*poslední znak předchozího odstavce * znak konce odstavce * první znak nového odstavce*

Protože ignorujeme „poslední“ a „první“ znaky, vyhledáváme a upravujeme pouze prostřední sekvenci, tj. konec odstavce a tabulátor, který se snažíme odstranit. To vše může vypadat složitě pouze do chvíle, než si vše ukážeme na příkladu. V používaném Wordu by nastavení vyhledávacího dialogu vypadalo takto:



7.VI: Vyhledávání speciálních znaků ve Wordu

Jediné, co je potřeba znát či umět dohledat, jsou metaznaky pro „tabulátor“ a „konec odstavce“. Ačkoliv jsou intuitivní (znaky zobrazené zde jsou platné i v mnoha jiných programech), není třeba je znát na zpaměť, jsou jednoduše dostupné v rámci samotného vyhledávacího dialogu (ve Wordu přes kombinaci Více → Speciální, v InDesignu přes znak rozšířené nabídky, viz obr. 7.VII)



7.VII: Vyhledávání speciálních znaků v InDesignu

Z přehledu samotných metaznaků je zřejmé, že funkce Vyhledat/Nahradit je silný nástroj, který může mnoho věcí zjednodušit. Platí to tím více, že vyhledávat můžeme také formáty znaků nebo jednotlivé styly, aniž by bylo nutné definovat hledaný text, je tedy možné globálně odstranit autorem používané podtržení textu a podobné nešvary.

Nejnovější verze InDesignu k těmto již dlouhodobě známým funkcím (ačkoliv je třeba dodat, že se objevily dříve u textových procesorů než u DTP programů) přidává další pokročilé prvky, mezi které patří práce s objekty a vyhledávání/nahrazování pomocí tzv. regulárních výrazů, schované v InDesignu pod záložku GREP. Vyhledávání pomocí regulárních výrazů patřilo dlouho do výbavy spíše specializovaných textových editorů nebo třeba databázových programů a může být na první pohled obtížné, má však své výhody, mezi které patří také možnost ukládat jednotlivá hledání a používat je i v jiných souborech. Navíc InDesign CS3 umožní uživateli poměrně jednoduchým způsobem odstranit další možný problém při sazbě, a to jsou jednoznakové neslabičné předložky na konci řádků: k tomu využijte funkci GREP s hodnotami

Vyhledat: (\\<[szkvaouiouSZKVAIOU]] []+

Nahradit: \$1~S

(Pro starší verze InDesignu bylo běžné využívat zdarma šířený plug-in model od distributora produktů Adobe v České republice, firmy Amos. Pro InDesign CS3 je však jednodušší využít výše uvedený postup.)

7.6 Makra a skriptování

Oba programy zmiňované výše umožňují také pokročilé prvky automatizace, mezi které patří vytváření maker a skriptů pomocí jazyka VisualBasic (produkty Microsoftu), respektive JavaScript (InDesign). Makra a skripty jsou vlastně specifické podprogramy, které jsou schopné běhu „uvnitř“ daného programu za účelem automatického zpracovávání některých úloh nebo úkonů. Makra, tak jak jsou často chápána, mají ještě jednu výhodu – program si je umí „naprogramovat“ sám podle toho, jak do programovacího jazyka interpretuje jednotlivé pracovní kroky uživatele. (Jinými slovy uživatel počítači jednou předvede sekvenci kliknutí, psaní, vyznačování a jiných úprav a program si tento postup zapamatuje a uloží jako makro pro další možné využití.) I makra je však samozřejmě možné psát přímo ve vybraném programovacím jazyce.

Psaní a využívání skriptů je samozřejmě daleko za zaměřením a rozsahem tohoto skriptu, chceme však uživatele upozornit, že na internetu lze nalézt množství volně stahovatelných, již hotových produktů. Mnohé z nich jsou zaměřeny právě na funkce, o kterých zde byla a je řeč – odstraňování špatně vložených spojovníků, kontrola a oprava uvozovek, kontrola osamocených neslabičných předložek a jednoznakových spojek apod. Bohužel, pokud je nám známo, neexistuje žádná „centrální databanka“ těchto nástrojů, a to ani takových, které by měly své využití v oblasti DTP. Desítky a desítky rozšíření (známá jako plug-ins nebo add-ons) pro jednotlivé produkty lze však velmi často najít na webech jejich výrobců, případně na specializovaných profesních webových serverech, z nichž některé jsou uvedeny v kapitole 8.

8 PŘÍLOHY

A. Doporučená literatura

- Bann, David: *Polygrafická příručka*. Praha 2008.
- Beran, Vladimír: *Typografický manuál*. Náchod 1994, Praha 1999², 2003³.
- Dančo, Václav: *Kapesní průvodce [počítačovou] typografií*. Via Vestra-Labyrinth, Praha, 1995.
- Dugong, Jean-Luc – Siegartová, Fabienne: *Typografie, od olova k počítačům*. Praha 1997.
- Eliška, Jiří: *Vizuální komunikace*. Písmo. Brno 2005.
- Grafická úprava tiskovin. Kolektiv autorů. SPN Praha 1990.
- Hlavsa, Oldřich: *Typographia I, II, III*. SNTL Praha 1976, 1981.
- Kočíčka, Pavel – Blažek, Filip: *Praktická typografie*. Brno 2000, 2004².
- Muzika, František: *Krásné písmo ve vývoji latinky I, II*. Reprint prvního vydání. Litomyšl 2005.
- Pistorius, Vladimír: *Jak se dělá kniha*. Praha – Litomyšl 2003, 2005².
- Sazba I. Kolektiv autorů. Praha 1984.
- Tůma, Tomáš: *Počítačová grafika a design. Průvodce začínajícího grafika*. Brno 2007.
- Vostrovský, Václav: *Základy počítačového publikování*. Praha 2003.
- Žára, Jiří – Beneš, Bedřich – Felkel, Petr: *Moderní počítačová grafika*. Brno 1998 (+ errata uveřejněná na <http://www.cgg.cvut.cz/ModerniPocitacovaGrafika/vydani1/errata.html>, 5. 6. 2008); tiž a Sochor, Jiří: *Moderní počítačová grafika*. Brno 2005².

B. Stránky s informacemi ze světa DTP, pre-pressu, grafiky atd.

- | | |
|---|--|
| http://www.grafika.cz | Grafika On-line. Odborný i obchodní web (Grafika Publishing). |
| http://www.printing.cz | Printing.cz. Server zaměřený na tisk, pre-press a knihařské zpracování. |
| http://www.typo.cz | „Česká a světová typografie na internetu“. Web pod kuratelou Filipa Blažka, spoluautora jedné z výše uvedených knih. |
| http://www.typomil.com | „Písmo, typografie a sazba na internetu“. Autorský web Martina Peciny. |
| http://www.sazba.cz | Komerční informace (Studio Lacerta) i všeobecné informace o sazbě, především v sekci Návody, texty, rady. |
| http://www.myfonts.com/ | Komplexní web o počítačových fontech. Součástí je zdarma dostupná služba, která umí dle naskenované ukázky rozpoznat použitý font. |
| http://www.unie-grafickeho-designu.cz/ | – Unie grafického designu. |
| http://www.cstug.cz/ | Československé sdružení uživatelů TeXu. |

C. Vybrané glyfy a jejich ASCII kódy

NB: Uváděné kódy zadávejte pomocí levého tlačítka ALT; nevynechávejte nulu před vlastním kódem.

„cze“ = ALT + 0132, ALT + 0147

,cze‘ = ALT + 0130, ALT + 0145

“eng” = ALT + 0147, ALT + 0148

‘eng’ = ALT + 0145, ALT + 0146

% = ALT + 037

& = ALT + 038

@ = ALT + 064

... = ALT + 0133

† = ALT + 0134

‰ = ALT + 0137

• = ALT + 0149

– = ALT + 0150

— = ALT + 0151

\$ = ALT + 0167

« = ALT + 0171

± = ALT + 0177

» = ALT + 0187

× = ALT + 0215

ä = ALT + 0228

ö = ALT + 0246

ü = ALT + 0252

D. Normalizované rozměry papíru

Rozměry jsou udávány v milimetrech.

Výchozí řada A se odvozuje od základního archu A0 o velikosti (plochy) 1 m². Řada B se odvozuje od archu B0, jehož šířka je 1 m. Řada C je propojovací, v praxi se používá často pro určení velikosti obálek atd. Řady B a C jsou tzv. doplňovací, A je základní.

Přestože jde o normalizované rozměry, doporučujeme v případě komplikovanějších tisků, tisků na spadávkou, tisků na specializované papíry atd. rozměry nejdříve konzultovat s tiskárnou, v níž se tisk bude realizovat, nebo s příslušným vydavatelstvím.

řada A

A0	841 × 1189
A1	595 × 841
A2	421 × 595
A3	297 × 421
A4	210 × 297
A5	149 × 210
A6	105 × 149

řada B

B0	1000 × 1414
B1	707 × 1000
B2	500 × 707
B3	354 × 500
B4	250 × 354
B5	177 × 250
B6	250 × 177

řada C

C0	917 × 1279
C1	648 × 917
C2	458 × 648
C3	324 × 458
C4	229 × 324
C5	162 × 229
C6	114 × 162

Vzhledem k zaokrouhlování se v některých příručkách tyto rozměry objevují s drobnou variací.

Tato publikace vychází v rámci grantového projektu

*Inovace studijních programů bohemistiky s ohledem
na potřeby soudobé informační společnosti.*

Tento projekt je spolufinancován Evropským sociálním fondem
a státním rozpočtem České republiky.

ÚVOD DO TYPOGRAFIE A POČÍTAČOVÉ GRAFIKY

Vladimír P. Polách

Obálka: Lenka Zuštinová – Algraf, Olomouc

Technický redaktor: Vladimír P. Polách

Odborný redaktor: PhDr. Jan Schneider, Ph.D.

Odpovědný redaktor:

Vydala Univerzita Palackého v Olomouci

Křížkovského 8, 771 47 Olomouc

www.upol.cz/vup

e-mail: vup@upol.cz

Olomouc 2008

1. vydání

ISBN