Sparing: Porównanie formatów Type 1 i TrueType

Adam Twardoch

Wstęp

Ostatnie dziesięciolecie było dla drukarstwa i typografii okresem rozwoju wprost rewolucyjnego. Słowo drukowane przestało być domeną zecerów – speców od czarnej roboty. „Piekielne komputerki” (jak mawia warszawski typograf, Stefan Szczypka) trafiły pod strzechy, zaś wypluwane z tanich „plujek” zaproszenia na urodziny ciotki Klotki zaczęły mienić się feerią kolorowych czcionek.

Również zawodowcy – redaktorzy, plastycy – zaczęli znów (po kilkuset latach zależności od właścicieli drukarń) mieć bezpośredni wpływ na ostateczny kształt drukowanego słowa.

Największe domy typograficzne, niegdysiejsze koncerny, od westchnienia których trząsł się świat, zaczęły przegrywać w konkurencji z małymi, prężnymi, czasem wręcz jednoosobowymi „odlewniami”, z dnia na dzień tworzącymi nowe kroje pisma. Czytelność (ang. *legibility*) została zastąpiona przez antyczytelność, realizm przez postmodernizm, typograficzną perfekcję zastąpiło mające wszystko usprawiedliwiać słowo *cool*.

W tym właśnie klimacie na typograficznej scenie przebiegał zacięty pojedynek dwóch wielkich koalicji. Pojedynek dwóch technologii, który miał zadecydować o tym, jak będzie wyglądał alfabet XXI wieku.

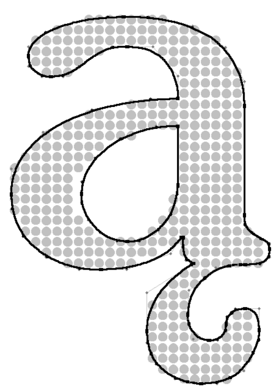
Font wagi piórkowej

Przez stulecia litery pojawiały się w sposób czysto mechaniczny – były wynikiem odbicia na papierze wizerunku odlanego w specjalnym stemplu nazywanym *czcionką*. Dziś miejsce czcionek zajął nowy nośnik pisma – *fonty*. Fonty komputerowe to zbiory danych cyfrowych – obwiedni tworzących poszczególne znaki, zdefiniowanych przy użyciu równań drugiego lub trzeciego stopnia, a także licznych danych dodatkowych. Fonty takie, zwane często wektorowymi, mogą być swobodnie skalowane, dzięki czemu uzyskać możemy wizerunek znaku o dowolnych rozmiarach.

Raster wynik nam pokaże

Proces tworzenia z fontu wizerunku litery zadanej wielkości nazywamy *rastrowaniem,* program, który za ten proces jest odpowiedzialny, nosi zaś nazwę *rasteryzator* (ang. *rasterizer*).

Rasteryzatory są zwykle integralnymi elementami systemu zarządzania fontami (ang. *type engine*), do którego zadań należy m.in. łączenie odmian pism w rodziny, tworzenie w oparciu o dany font odmian pochodnych, dostęp do znaków specyficznych dla danego języka itp. Choć jest to nieprecyzyjne, często spotyka się zamiennie użycie nazw *font rasterizer* i *type engine*.



Rys. 1. Rastrowanie fontu

Rasteryzatory fontów, a wraz z nim i *type engines*, mogą być częścią systemu operacyjnego (jak np. rasteryzator fontów TrueType w MS Windows) lub występować jako nakładki na system (np. Adobe Type Manager).

Rastrowanie polega na wypełnianiu kropkami kształtu zdefiniowanego obwiednią. Kropki mogą być albo ekranowymi pikselami, albo kropelkami farby drukarskiej czy okruszkami tonera.

Na danym wycinku powierzchni „zadruku” można zmieścić ściśle określoną liczbę tychże kropek. Zależy to od tego, jakie materiały („farba” i podłoże, którym oczywiście może być luminofor ekranu monitora) użyte są w tym procesie, oraz od tego jaką zastosowano technikę druku czy wyświetlania. Maksymalną liczbę kropek, jaką można przy użyciu konkretnego urządzenia (np. monitora, drukarki, naświetlarki, maszyny drukarskiej) zmieścić na zadanym odcinku należącym do powierzchni „zadruku”, nazywamy *rozdzielczością* tego urządzenia. Oczywiście im wyższa rozdzielczość urządzenia, tym lepiej.

Przykładowo, rozdzielczość ekranu waha się w granicach 72 – 96 dpi (punktów na cal), podczas gdy papier zadrukowywany jest w rozdzielczościach od 300 dpi w przypadku drukarek do kilku tysięcy punktów na cal w przypadku profesjonalnych maszyn drukarskich.

Najpopularniejszymi stosowanymi dziś formatami są formaty *Type 1* i *TrueType*.

Przed pojedynkiem

Format *Type 1* opracowany został w roku 1985 [1] z myślą o wykorzystaniu w urządzeniach drukujących posługujących się językiem opisu strony *PostScript*. Pomysłodawcą przedsięwzięcia była firma Adobe Systems, Inc. W tym czasie na rynku pojawiły się już pierwsze komputery Apple, w których zastosowane zostało graficzne środowisko, oparte na technologii wyświetlania obrazu zwanej *QuickDraw* i drukarki laserowe umożliwiające znakomitą jak na tamte czasy jakość druku w rozdzielczości 300 dpi. Firma Apple, zamierzająca wyprodukować własną drukarkę laserową przeznaczoną dla swoich komputerów, rozpoczęła poszukiwania możliwie uniwersalnego języka opisu strony, a więc odpowiednika QuickDraw, z tym, że przeznaczonego dla drukarek.

Próby stworzenia takiego języka podjęło kilka firm, spośród których Apple wybrał opracowany przez Adobe, rewolucyjny jak na tamte czasy, język PostScript, gwarantujący (przynajmniej w teorii) możliwość niezależnego od typu urządzenia końcowego przygotowania danych do druku. Apple wyposażył w ten język swoją pierwszą drukarkę – Apple LaserWriter. Dzięki swym licznym zaletom, PostScript błyskawicznie zaczął stawać się bardzo popularny również w kręgach profesjonalnych, wypierając specyficzne rozwiązania poszczególnych producentów. Gdy okazało się, że ogromna większość firm produkujących urządzenia drukarskie (drukarki i naświetlarki) zapragnęła wyposażyć swoje maszyny w PostScript, Adobe zaczęło domagać się wysokich opłat licencyjnych za interpreter PostScriptu, zawierający m.in. rasteryzator krojów Type 1.

Specyfikacja formatu Type 1 trzymana była w tajemnicy, głównie ze względu właśnie na licencje, a także na opracowany przez Adobe mechanizm optymalizacji kształtu liter w niskich rozdzielczościach, tzw. *hinting*. Opublikowano tylko specyfikację formatu *Type 3*, który pozbawiony był tego cennego mechanizmu.

Wielcy producenci oprogramowania (Apple, IBM, Microsoft) zaczęli sobie jednak szybko zdawać sprawę, że warto by wykorzystać fonty skalowalne również w tworzonych przez nich systemach operacyjnych – do tej pory znaki na ekranie wyświetlane były w oparciu o matryce zapisane w postaci map bitowych.

Ponieważ specyfikacja formatu Type 1 nie była ogólnie dostępna, producenci oprogramowania, chcąc wykorzystać ten format w swoich systemach, musieliby uzależnić się od Adobe, i opłacić horrendalne licencje. W roku 1987 firmy Apple i Microsoft zdecydowały więc opracować własny system wyświetlania grafiki i nowy format zapisu fontów, który między innymi w większym stopniu uwzględnić miał specyfikę wyświetlania znaków na ekranie.

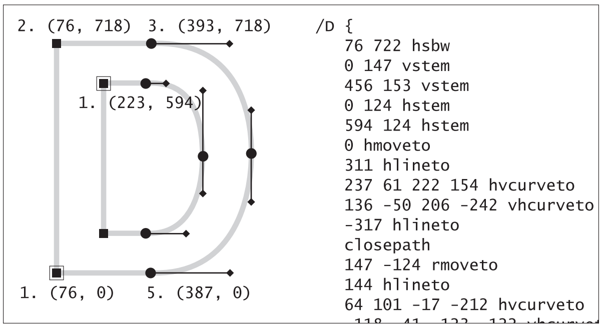
W ramach tego porozumienia Microsoft rozpoczął prace nad opracowaniem podobnego zasadą działania do PostScriptu języka opisu strony o nazwie *TrueImage*, Apple miał zaś stworzyć nowy format zapisu fontów, któremu nadano tymczasową nazwę *Royal*. Prace Microsoftu nad TrueImage nie przyniosły oczekiwanych rezultatów, a więc wkrótce je zaprzestano. Jednak informatycy Apple, na czele których stał *Sampo Kaasila*, po dwóch latach prac stworzyli *TrueType*.

Aby ratować pozycję Type 1, firma Adobe odtajniła w roku 1990 specyfikację tego formatu, i na początku 1991 roku stworzyła *Adobe Type Manager (ATM)*, rasteryzator (oraz *type engine*) fontów Type 1 dla komputerów Apple.

W tym samym okresie, w marcu 1991 roku, Apple zaprezentował format TrueType. Okazało się, że format ten dużo lepiej od Type 1 uwzględniał specyfikę wyświetlania znaków na ekranie – miał dużo lepszy mechanizm hintingu i znacznie szybszy rasteryzator. Firma Apple zakupiła więc od amerykańskiego koncernu poligraficznego *Linotype* licencje na wykorzystanie w swoim systemie operacyjnym krojów *Times Roman*, *Helvetica* i *Courier*, które już wcześniej znalazły się w repertuarze fontów dołączanych przez Adobe do interpretera PostScriptu. Inżynierowie firmy Apple stworzyli ich znakomite wersje TrueType, które reprezentacją na ekranie biły na głowę postscriptowe odpowiedniki.

Rok później technologia TrueType miała swoją drugą premierę – tym razem wchodząc w skład *Windows 3.1* – nowej wersji systemu operacyjnego firmy Microsoft, która – we współpracy z drugim poligraficznym gigantem, *Monotype* – przygotowała truetypowe wersje krojów *Arial*, *Times New Roman* i *Courier New*, zbliżonych do „oryginałów” zastosowanych w PostScripcie, jednak nieznacznie różniących się ze względów licencyjnych [2].

Niestety okazało się, że 16-bitowy rasteryzator Microsoftu miał sporo błędów. Również implementacja hintingu nie została dokonana poprawnie.



Rys. 2. Budowa znaku w foncie Type 1

W związku z tym większość poważnych producentów pism drukarskich, na czele z Linotype, nie zdecydowała się na przeniesienie swoich krojów do formatu TrueType i zaczęła oferować je tylko w postaci fontów Type 1.

Ring wolny

Type 1. *Type 1* [3] składa się z dwóch plików: metrycznego i binarnego. Plik binarny (zakodowany, zwykle nosi rozszerzenie `.pfb`) ma budowę słownikową. Istnieje kilka „słowników" zawierających dane o kroju pisma – przede wszystkim o kształtach liter i rozmieszczeniu hintów.

Znaki w fontach Type 1 zbudowane są z obwiedni, opisanych tzw. *krzywymi Béziera trzeciego stopnia*. Krzywa taka definiowana jest przez ciąg *punktów korpometrycznych (punktów kontrolnych, węzłów*, ang. *control points, nodes)* ułożonych w dwuwymiarowym układzie współrzędnych, na obszarze *firetu* (ang. *em square*).

Plik metryczny (o rozszerzeniu `.afm`) zawiera informacje o rozmiarach znaków oraz kerningu (korekta optyczna świateł międzyliterowych).

Fonty Type 1 mogą zawierać dowolną liczbę znaków, jednak dla większości aplikacji w najpopularniejszych systemach operacyjnych dostępnych jest ich maksymalnie 256.

Jest to znaczne ograniczenie, praktycznie uniemożliwiające obsługę wielu języków czy stosowanie ligatur lub form alternatywnych w jednym foncie. Jednym ze sposobów ominięcia tej bariery jest umieszczenie znaków danej odmiany w kilku fontach – tworzy się wówczas tzw. *expert sets*, zawierające kapitaliki, ligatury i formy alternatywne, albo fonty zakodowane według różnych *stron kodowych*, zawierające znaki różnych alfabetów. Istnieje co prawda rozszerzenie formatu Type 1 pod nazwą *CID-keyed fonts* [4], stworzone przez firmę Adobe na potrzeby rynków azjatyckich, które umożliwia stosowanie fontów zawierających kilka czy nawet kilkadziesiąt tysięcy znaków, jednak definicja tego formatu jest zawiła, a i tworzenie fontów CID jest skomplikowane.

Ciekawostką jest, że w formacie *Type 3*, który nie zawiera informacji o hintach, znaki mogą być kreślone przy użyciu pełnej palety instrukcji postscriptowych. Umożliwia to stosowanie rozmaitych „efektów specjalnych". Np. w piśmie *FF Beowolf* [8] firmy *LettError* (format Type 3) wszystkie znaki podlegają w czasie drukowania ciągłej niewielkiej losowej modyfikacji, dzięki czemu ta sama litera za każdym razem przybiera minimalnie inny kształt.

TrueType. Format *TrueType* [6] jest od formatu Type 1 dużo bardziej skomplikowany, posiada za to więcej możliwości i nietrudno go rozszerzać zachowując kompatybilność. Fonty TrueType, zawarte w jednym pliku, zbudowane są w postaci serii rekurencyjnych tabel, z których każda opisuje inny aspekt fontu.

Obwiednie znaków w fontach TrueType zapisane są w tabeli *glyf* przy *pomocy krzywych Béziera drugiego stopnia*, tzw. *b-splines*. Dzięki prostszej matematycznej formule fonty TrueType są szybciej kreślone na ekranie. Aby wydrukować taki font w urządzeniu postsciptowym, do niedawna system operacyjny musiał konwertować fonty TrueType do postaci Type 1, co nie zawsze odbywało się zgodnie z oczekiwaniami – czasem pojawiały się błędy, wynikające głównie z zaokrągleń przy konwersji współrzędnych, które oparte są na różnych skalach. Stąd utarło się przekonanie, że fonty TrueType nie nadają się do profesjonalnego przygotowywania publikacji, a tylko do zastosowań biurowych. Na opinię tę wpłynęła również fatalna jakość rasteryzatora TrueType dołączonego do Microsoft Windows 3.1.

Na szczęście praktycznie wszystkie nowe postscriptowe urządzenia drukarskie „rozumieją" także TrueType (wysyłane do drukarki w specjalnym formacie Type 42), zaś rasteryzator TrueType zawarty w Windows 95 i NT 4.0 jest znacznie poprawiony, a więc problem już nie istnieje.

Wraz z wprowadzeniem przez Microsoft w 1993 roku systemu Windows NT 3.1, fonty TrueType zaczęły obsługiwać przyszłościowy standard kodowania znaków *Unicode* [7], będący częścią

Rys. 3. Przykład kilku odmian kroju Adobe Myriad MM (w technologii Multiple Master)

Rys. 4. Tekst zrastrowany w różnych rozdzielczościach

międzynarodowej normy *ISO 10646*. Jest w nim zdefiniowanych (w tabeli *cmap*) kilkadziesiąt tysięcy znaków spotykanych w niemal wszystkich alfabetach świata. Są oczywiście również polskie znaki diakrytyczne.

Dwa lata później Microsoft przedstawił nową wersję swojego systemu: Windows 95. System ten zawierał znacznie ulepszony rasteryzator TrueType (dodano możliwość wygładzania krawędzi liter), a także mechanizm zwany *Windows Glyph List 4 (WGL4),* umożliwiający m.in. połączenie Unicode z dotychczasowymi ośmiobitowymi stronami kodowymi. Dzięki temu mechanizmowi nawet programy nie obsługujące Unicode mogą korzystać z fontów TrueType zakodowanych unikodowo, choć stosowanie tego mechanizmu jest mocno uciążliwe.

Dane metryczne i kerning fontów TrueType zapisane są w kilku osobnych tabelach, istnieje też sporo tabel dodatkowych, opisujących hinting, anti-aliasing i inne aspekty techniczne, a także zawierających liczne informacje na temat nazwy kroju pisma, projektanta, stanu autorskoprawnego i licencyjnego fontu itp.

Rozszerzenia obu formatów. Poza krojami postscriptowymi (Type 1) i krojami TrueType, istnieją również rozszerzenia obu formatów.

Jak wiadomo, poza popularnym Type 1, firma Adobe opracowała kilka innych formatów zapisu fontów (jak wspomniane już Type 3 czy Type 42).

Jednym z tych formatów jest format *Multiple Master (MM)*, umożliwiający dynamiczne tworzenie odmian danego kroju na drodze interpolowania wzdłuż jakiejś osi między zadanymi odmianami ekstremalnymi. Osiami interpolacji mogą być m.in. grubość, szerokość, wielkość optyczna (przypomina to nieco koncepcję Metafonta).

Kroje Multiple Master tworzone są w zasadzie jedynie przez projektantów z firmy Adobe, wśród których przoduje *Robert Slimbach*. Żaden z tych krojów nie posiada polskich znaków, więc ich przydatność w naszych warunkach jest znikoma.

Mało znanym, a bardzo ciekawym rozszerzeniem systemu QuickDraw, odpowiedzialnego za wyświetlanie obrazu na komputerach Apple, jest *QuickDraw GX*. Jest to rozszerzenie zarówno formatu TrueType jak i Type 1, dostępne jednak tylko na komputerach Apple Macintosh.

Kroje GX mają możliwości podobne do krojów Multiple Master, a ponadto umożliwiają dostęp do więcej niż 256 znaków w foncie. Posiadają też mechanizmy obsługi ligatur, znaków alternatywnych i specjalnych. Technologia ta nie zdobyła jednak nigdy większej popularności, szczególnie ze względu na bardzo zawile napisaną specyfikację, a także z powodu dostępności tylko na platformie Apple.

Można przypuszczać, że rozszerzenia te stracą zupełnie na ważności z momentem upowszechnienia formatu OpenType.

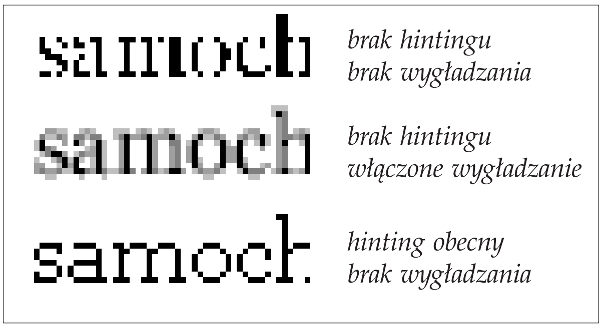
Hinting — lewy sierpowy

Jeżeli niewielkich rozmiarów obiekt, np. literę, wykreślimy na urządzeniu o niskiej rozdzielczości (np. ekran monitora), to może się okazać, że podczas rastrowania utracone zostaną liczne szczegóły.

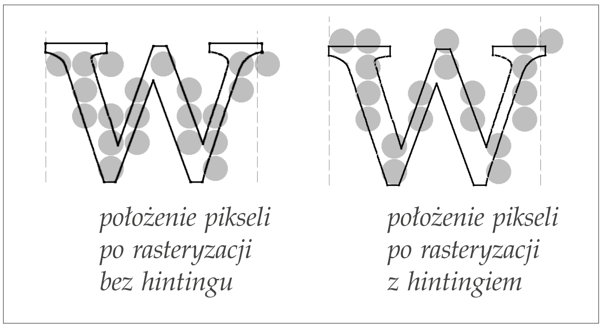
Typowy przykład: jeżeli jedna z nóżek litery n ma 149 jednostki szerokości, zaś druga 150, to po zrastrowaniu litery w rozdzielczości ekranu może się okazać, że w wyniku zaokrąglania jedna z nóżek ma 1 piksel grubości, zaś druga 2 piksele, co jest różnicą bardzo znaczącą. Jeszcze większe problemy pojawiają się przy rastrowaniu elementów ukośnych i łuków.

Aby tych problemów uniknąć, producenci fontów stosują rozmaite techniki polepszające czytelność tekstów na ekranie. Do najpopularniejszych należą: *hinting* [8], wygładzanie krawędzi (ang. *anti-aliasing*) [9], a także inne, np. wprowadzona niedawno (w połowie listopada 1998) przez firmę Microsoft technologia *ClearType* [10], która znacznie ulepsza wyświetlanie tekstu na ekranach ciekłokrystalicznych.

Pozornie zdawać by się mogło, że techniki te nie są potrzebne, gdy fontu używa się tylko do



Rys. 5. Hinting i wygładzanie krawędzi



Rys. 6. Położenie pikseli po rasteryzacji z hintingiem i bez hintingu

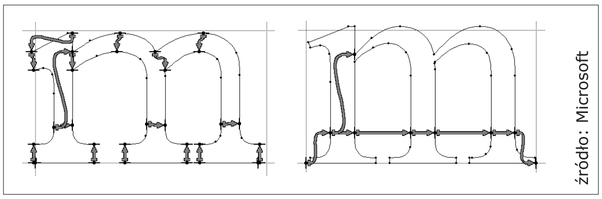
tworzenia dokumentów papierowych. Pamiętać jednak trzeba, że w zasadzie każdy font jest oglądany na ekranie – choćby przez operatora w czasie przygotowywania publikacji do druku.

Jak widać na rysunku , wygładzanie krawędzi poprawia czytelność tekstu, jednakże dużo większy wpływ na dobre prezentowanie się pisma w niskich rozdzielczościach ma *hinting*.

Instrukcje hintujące wykonywane są przez rasteryzator, który odczytuje zapisane w foncie wskazówki (ang. *hints*), i kierując się nimi zmienia położenie poszczególnych kropek tworzących obraz litery, by zminimalizować niedokładności rastrowania.

Pomiędzy technologiami hintingu stosowanymi w fontach Type 1 i TrueType istnieją zasadnicze różnice.

Hinting w Type 1. Hinting w fontach Type 1 jest zbiorem „słów-wskazówek”, które przypisywane są poszczególnym pionowym i poziomym elementom różnych liter. Rozmiar wszystkich elementów opatrzonych tym samym „słowem” powinien być w przypadkach wątpliwych zaokrąglany to tej samej liczby pikseli (por. *hstem* i *vstem* na rys.). Decyzje o włączaniu i wyłączaniu pojedynczych pikseli podejmuje rasteryzator (ATM),



Rys. 7. Instrukcje hintujące TrueType

który – co za tym idzie – musi być wyposażony w dość skomplikowany system analityczny.

Taki rodzaj hintowania nie pozwala projektantowi na dokładną kontrolę zachowania fontu na ekranie, a także znacznie spowalnia proces wyświetlania na ekranie. Hinting Type 1 nie rozwiązuje też problemów z rastrowaniem elementów ukośnych i okrągłych. Wymaga jednak od projektanta stosunkowo niewielkiego nakładu pracy.

Jak już wspomniałem wcześniej, na jakość percepcji tekstu wyświetlanego na ekranie pozytywnie wpływa też wygładzanie krawędzi (anti-aliasing). Program Adobe Type Manager od wersji 4.0 umożliwia anti-aliasing na ekranie, jednak włączenie tej opcji znacznie spowalnia proces wyświetlania tekstu.

Hinting w TrueType. W fontach TrueType można przy pomocy hintów kontrolować ze stuprocentową dokładnością każdy pojedynczy piksel pojawiający się na ekranie w każdej rozdzielczości. Innymi słowy, hinting TrueType jest wykonywany przez rasteryzator dokładnie tak, jak to zaplanował projektant fontu. Dzieje się tak za sprawą specjalnych instrukcji.

Instrukcje hintujące są opisane na dwóch poziomach – ogólnym i szczegółowym. Zasada działania hintów *ogólnych* jest nieco zbliżona do hintów Type 1. Projektant fontu ustala zależności pomiędzy poszczególnymi punktami kontrolnymi: może np. ustalić, że środkowa nóżka litery „m” powinna zawsze znajdować się dokładnie pośrodku nóżek skrajnych. Zestaw instrukcji hintujących TrueType jest jednak znacznie bogatszy niż w przypadku Type 1. Istnieją specjalne instrukcje hintujące elementy ukośne, przez co widocznie lepiej wyświetlane są truetypowe *kursywy*.

Poziom *szczegółowy* hintingu jest to zestaw programów definiowanych osobno dla każdego stopnia pisma w określonej rozdzielczości, mierzonej w pikselach na firet (*ppm* – *pixel per em*). Tzw. instrukcje *delta* pozwalają włączać i wyłączać poszczególne piksele. Dzięki temu można osiągnąć efekty takie, jakie pokazano na rys. i .

Hintowanie TrueType wymaga znacznie większego nakładu pracy ze strony projektanta – za to efekty są znacznie lepsze niż w przypadku hintów Type 1. Technologia hintingu stosowana w fontach TrueType ma też tę zaletę, że nie wymaga od rasteryzatora żadnych dodatkowych obliczeń, co znacznie przyspiesza proces rastrowania.

Oznacza to, że tak doskonałą czytelność liter na ekranie w małych stopniach pisma, jaką osiągnięto w przypadku dostępnych bezpłatnie, nowych fontów firmy Microsoft [12] (*Verdana*, *Georgia*, *Trebuchet*), osiągnąć można tylko stosując TrueType. Trzeba jednak podkreślić, że tak znakomite efekty zrealizować można tylko znacznym nakładem finansowym, angażując programistę profesjonalnie zajmującego się hintingiem.

Podobnie jak Adobe, również Microsoft wprowadził do swojego rasteryzatora wygładzanie krawędzi fontów. Anti-aliasing odbywa się przy użyciu czterech (w Windows 95/98) lub siedemnastu (w Windows NT) odcieni tonalnych, przy czym projektant fontu TrueType może (poprzez odpowiednie ustawienia w tabeli fontu *gasp*) precyzyjnie zdefiniować, w których stopniach pisma font ma być hintowany oraz wygładzany. Jest to o tyle istotne, że w wielu przypadkach (szczególnie w bardzo małych rozmiarach pisma, poniżej 16 ppem) niecelowe jest jednoczesne stosowanie hintingu i wygładzania, gdyż może to nawet ujemnie wpłynąć na czytelność tekstu.

Kto wygrał?

Type 1 i TrueType. Niewątpliwą przewagą formatu TrueType nad Type 1 jest uniwersalność tego pierwszego – nadaje się on zarówno do definicji fontów „dużych" (zawierających znaki języków azjatyckich), jak i „małych" (zawierających litery alfabetów europejskich).

Format TrueType jest także w pełni skalowalny (rozszerzalny), co oznacza, że nowo opracowane fonty, zawierające np. znaki Unicode, działają poprawnie również na starych urządzeniach i systemach (w zakresie jaki dopuszcza „stary" rasteryzator). Właśnie z tego względu na strukturze plików TrueType oparto nowy format OpenType.

Format Type 1 ma jednak również sporo niepodważalnych zalet. Jego budowa jest przede wszystkim prosta i przejrzysta, co umożliwia tworzenie dużo mniejszym nakładem środków zarówno fontów w tym formacie, jak i narzędzi do obróbki tychże. Ponadto specyfikacja formatu Type 1 napisana jest staranniej i przejrzyściej niż specyfikacja TrueType, którą Bogusław Jackowski skomentował [14] słowami: „jest to najgorsza, najbardziej mętna dokumentacja jaką zdarzyło mi się trzymać w rękach; w grę wchodzi jedynie wzięcie dokumentacji do ręki – o czytaniu mowy nie ma."

Format krzywych, z których konstruowane są obwiednie znaków, zgodny jest z ogólnie przyjętym standardem, stosowanym w języku PostScript i w wielu programach ilustracyjnych.

Co do hintingu, to twórca fontów Type 1 może szybko osiągnąć przyzwoity efekt, ale nic ponadto, z drugiej zaś strony przygotowując font TrueType można stworzyć dzieło wprost znakomite, jednak nieproporcjonalnie większym nakładem pracy.

OpenType. Najnowszym formatem fontów jest *OpenType* [15]. Inicjatywa stworzenia go powstała w 1996 roku, kiedy Adobe i Microsoft podpisały porozumienie o zakończeniu „wojny fontowej" i planach wprowadzenia jednego, wspólnego formatu opisu fontów, łączącego zalety obydwu.

Format plików OpenType jest rozszerzeniem formatu TrueType, dzięki czemu fonty OpenType mogą być (w ograniczonym, rzecz jasna, zakresie) stosowane w starszych systemach operacyjnych.

Nowością w OpenType jest możliwość zastąpienia truetypowej tabeli *glyf* tabelą *CFF*, w której – po automatycznej konwersji do wewnętrznego formatu *CFF/Type 2* – umieszczone mogą być postscriptowe obrysy fontu Type 1. Dzięki temu zniknie konieczność tworzenia i przechowywania fontów w obydwu formatach oraz konwertowania między nimi.

Dla zachowania kompatybilności w dół, pliki OpenType zawierające tylko obrysy TrueType opatrzone są rozszerzeniem `.ttf`, zaś pliki posiadające tylko obrysy Type 1 – `.otf`. Wykonanie plików posiadających obrysy obydwu typów jest co prawda technicznie możliwe, jednak obie firmy stanowczo to odradzają [16].

Podstawowym zestawem znaków stosowanym w OpenType jest Unicode, co pozwoli zakończyć trwający od wielu lat bałagan związany ze stosowaniem wielu stron kodowych. Naturalnie tworzenie

Rys. 8. Ligatury w OpenType

fontów OpenType z niestandardowymi wektorami kodowania nadal będzie możliwe.

Kolejną nowością, jaką przynosi ze sobą nowy format, są tzw. *OpenType Layout features*. Chodzi tu o zestaw predefiniowanych funkcji, których wykonanie będzie mógł fontowi zlecić program lub użytkownik. Wśród dotychczas zarejestrowanych ponad stu funkcji znajdują się trzy różne warianty wstawiania ligatur, zastępowanie zwykłego zera zerem kreślonym, wstawianie indeksów górnych i dolnych, zamiana tekstu na kapitaliki lub cyfr zwykłych na nautyczne (ang. *old style figures*), korekta optycznych nawiasów w tekście napisanym wersalikami, różne sposoby wyrównywania odstępów międzyliterowych itp.

Jeżeli dany font będzie posiadał asortyment znaków pozwalający na wykonanie odpowiedniej funkcji, program będzie mógł ją uaktywnić automatycznie lub na życzenie użytkownika, podobnie jak dziś włączane jest podkreślenie czy wybierana kursywa. Wszystkie te funkcje (oznaczane czteroliterowymi akronimami, np. *hlig*) mogą być stosowane w zależności od kontekstu – np. od alfabetu i języka, którymi dany tekst jest napisany.

Dzięki temu będzie można, bez ingerencji użytkownika, w dwujęzycznym tekście złożonym jednym krojem, stosować poprawne francuskie „o” z akcentem *acute* wraz z poprawym typograficznie polskim „o” z kreską (w którym akcent ma inny niż w języku francuskim kształt i położenie), pomimo iż obydwa te znaki występują w zestawie Unicode pod pozycją `0x00F3` (jako *oacute*).

Obsługa OpenType planowana jest w systemach operacyjnych MacOS 9 i Windows 2000 (NT 5.0), które mają ujrzeć światło dzienne pod koniec 1999 roku.

Może wówczas skończy się ten sparing?

Literatura

1. Laurence Penney, *A History of TrueType*, `http://www.truetype.demon.co.uk/tthist.htm`
   David K. Every, *What is the history of TrueType*, `http://www.mackido.com/History/History_TT.html`
   Thomas W. Phinney, *TrueType & PostScript Type 1: What's the Difference?*, `http://www.truetype.demon.co.uk/articles/ttvst1.htm`
2. Charles Bigelow, *Times (New) Roman and its part in the Development of Scalable Font Technology*, `http://www.truetype.demon.co.uk/articles/times.htm`
3. Adobe Systems, Inc., *Adobe Type 1 Font Format*, Addison-Wesley, 1990, ISBN 0-201-57044-0, `http://partners.adobe.com/supportservice/devrelations/typeforum/ftypes.html`
4. Adobe Systems, Inc., *CID-Keyed Fonts*, `http://partners.adobe.com/supportservice/devrelations/typeforum/cidfonts.html`
5. Erik van Blokland, Just van Rossum, *FF Beowolf*, `http://www.letterror.com/LTR_Random.html`
6. Microsoft Typography Group, *TrueType specificaton v1.66*, `http://www.microsoft.com/typography/tt/tt.htm`
   Apple Computer, Inc., *TrueType Reference Manual*, `http://fonts.apple.com/TTRefMan/`
7. Unicode Consortium, `http://www.unicode.org/`
8. Microsoft Typography Group, *Introduction to hinting*, `http://www.microsoft.com/typography/hinting/hinting.htm`
   Beat Stamm, *The raster tragedy at low resolution*, `http://www.microsoft.com/typography/tools/trtalr.htm`
   Microsoft Typography Group, *How does TrueType hinting differ from other outline font formats?*, `http://www.microsoft.com/typography/hinting/versus.htm`
   Laurence Penney, *TrueType Hinting*, `http://www.truetype.demon.co.uk/tthints.htm`
9. Laurence Penney, *TrueType and Anti-Aliasing*, `http://www.truetype.demon.co.uk/ttalias.htm`
10. Microsoft Typography Group, *ClearType*, `http://www.microsoft.com/typography/cleartype/`
11. *FreeType*, `http://www.freetype.org/`
12. Microsoft Typography Group, *TrueType core fonts for the Web*, `http://www.eu.microsoft.com/typography/fontpack/`

13. Tom Rickner, *Get the Hint*, `http://www.will-harris.com/msfont-hint.htm`
14. Bogusław Jackowski, *Znaki na niebie, ziemi i gdzie indziej*, referat wygłoszony na Jesiennych Spotkaniach PTI, Mrągowo, 16–20. 11. 1998
15. Adobe Systems, Inc., Microsoft Corporation, *OpenType Specification v1.2*, `http://www.eu.microsoft.com/typography/OTSPEC/`

    Microsoft Typography Group, *Developer Information*, `http://www.eu.microsoft.com/typography/creators.htm`

    Adobe Systems, Inc., *The Adobe Type Technology Forum*, `http://partners.adobe.com/supportservice/devrelations/typeforum/`
16. Adobe Systems, Inc., Microsoft Corporation, *Recommendations for OpenType Fonts*, `http://partners.adobe.com/supportservice/devrelations/opentype/recom.htm`

◇ Adam Twardoch
`twardoch@font.org`

Konferencje

MultiTypo '98. Konferencja ATypI, Lyon, 23–25 października 1998

Adam Twardoch

W dniach od 23 do 25 października we francuskim mieście Lyon odbyła się konferencja MultiTypo '98, zorganizowana w cyklu dorocznych spotkań członków ATypI (Association Typographique Internationale tj. Międzynarodowego Stowarzyszenia Typograficznego). Od piątku do niedzieli w dwóch salach prelekcyjnych i kilku mniejszych pomieszczeniach wystawowych i dyskusyjnych nowoczesnego i bardzo okazałego Pałacu Kongresowego (Palais de Congrès), ponad 370 literników, grafików, księgarzy, edytorów i ludzi związanych z branżą wydawniczo-poligraficzną z całego świata (w tym czworo z Polski) dyskutowało na temat dokonań typografii francuskiej w przeszłości i obecnie, jak też najnowszych doniesień i trendów w typografii na świecie.

Gospodarze konferencji, Sharon Irving z brytyjskiego biura ATypI i młody (27-letni), świetnie zapowiadający się francuski typograf, Jean-François Porchez, przygotowali znakomity program.

Głównym motorem konferencji były – odbywające się równolegle w dwóch salach – prelekcje w języku angielskim i francuskim, tłumaczone symultanicznie.

Konferencji towarzyszyły też wystawy prac typograficznych z ostatniego roku, odbywające się równolegle do wykładów warsztaty, giełda książek traktujących o grafice i typografii, a także aukcja prac uczestników konferencji. Odbyło się także walne zgromadzenie członków Stowarzyszenia.

Piątek, 23 października. Pierwszy dzień konferencji wypełniony był nieformalnymi spotkaniami, dając uczestnikom czas na poznanie się i rozejrzenie „w terenie”. Uczestnicy konferencji mogli też wziąć udział w zwiedzaniu Muzeum Drukarstwa.

Późnym południem rozpoczęły się wykłady. Organizatorzy popełnili drobny błąd, umieszczając w dużej sali plenarnej „On Typo” wykład Philippe'a Millot na temat historii sztuki i jej wpływu na typografię. Wykład, z którym czasowo konkurowała prezentacja nowego, opracowanego przez Adobe i Microsoft formatu fontów OpenType. Prezentację prowadził David Lemon, kierujący oddziałem typografii w firmie Adobe, zaś sekundował mu Vincent Connaire z firmy Microsoft.

Stosunkowo niewielka sala „Off Typo” wypełniona była po brzegi. Lemon prezentował możliwości nowego formatu – ligatury, cyfry nautyczne, Unicode, skład pionowy. Pokaz przyjęto z zachwytem, zaś pytaniom nie było końca. David Lemon zachęcił do przeczytania specyfikacji formatu w wersji 1.2, która w tym samym czasie pojawiła się na internetowych stronach obu firm.

Sobota, 24 października. Sobotni poranek to przede wszystkim niezwykle ciekawy wykład Matthew Cartera z firmy Carter & Cone, autora m.in. pism ITC Galliard, Bell Centennial, Verdana, Georgia. Słynny typograf mówił o wpływie francuskiej typografii ubiegłych stuleci na obecne projekty pism drukarskich, m.in. na jego własny alfabet Galliard.

Wczesnym popołudniem najważniejsze bodaj wydarzenie konferencji – obchodzący kilka tygodni wcześniej swoje 70. urodziny szwajcarski mistrz typografii Adrian Frutiger w panelowej rozmowie z Yvonne Schwemer-Scheddin z monachijskiego Towarzystwa Typograficznego.

Adrian Frutiger – wspaniały, unikający zaszczytów, głęboko skromny „zwykły rzemieślnik” (jak sam o sobie mawia) to jeden z najwybitniejszych, a w mojej opinii najwybitniejszy typograf naszego stulecia, twórca kilkunastu alfabetów z Universem i Frutigerem na czele. W Lyon zachwycił on publiczność swymi gawędziarskimi rozważaniami z pogranicza filozofii, semiotyki, semiografii i typografii do tego stopnia, że musiano odwołać nie tylko przerwę na kawę, ale i pół następnego wykładu.

Pod wieczór nową (na razie testową) wersję znakomitego edytora fontów FontLab 4.0, wyposażoną m.in. w potężny język makropoleceń, zaprezentował Jurij Jarmola z Sankt Petersburga. FontLab, dostępny obecnie na platformie PC, wkrótce zaś dostępny też dla użytkowników Macintoshy, to drugi – obok pakietu Macromedia Fontographer – profesjonalny edytor fontów znajdujący się w sprzedaży. Umożliwia on operacje dostępne dotychczas jedynie użytkownikom specjalnych, wewnętrznych systemów edycji fontów stworzonych na potrzeby wielkich korporacji – Adobe, Monotype, Linotype czy Bitstream. Za pomocą programu FontLab można przygotowywać nie tylko wielojęzyczne fonty Type 1 i TrueType, ale także fonty Multiple Master czy CID. Dostępny jest m.in. wizualny hinting TrueType, wygodny edytor kerningu, moduł kontroli poprawności technicznej fontu FontAudit, funkcja automatycznej wektoryzacji map bitowych, a także liczne transformacje znaków w rodzaju dodania obrysu czy pogrubienia litery.

Niedziela, 25 października. Niedzielnym rankiem odbyło się kilka spotkań roboczych, dotyczących m.in. problematyki praw autorskich oraz kierunków rozwoju ATypI.

Około południa John Hudson z kanadyjskiej firmy Tiro Typeworks wspólnie z Vincentem Connaire z Microsoftu przedstawił pierwszą wersję specjalnej bazy danych, mającej być źródłem informacji na temat języków i alfabetów świata. Bazę ta firma Microsoft zamierza umieścić w Internecie. Specjalnie na tę okazję John Hudson z Tiro i Geraldine Wade z Monotype stworzyli w formacie OpenType nowy krój pisma Sylfaen, zawierający znaki większości alfabetów świata. Warto podkreślić, że alfabet cyrylicy kroju Sylfaen został wyróżniony w zorganizowanym w grudniu ub.r. w Moskwie konkursie nowych pism drukarskich Kyrillitsa 99 , zaś kształty polskich znaków diakrytycznych konsultowane były m.in. z piszącym te słowa.

Po południu bardzo ciekawe wykłady poprowadzili: Gerard Unger, który analizował historię francuskiej typografii na przykładach prac Roberta Granjona i Françoisa Didota oraz Liubow Kuzniecowa z moskiewskiej firmy ParaType, która wraz z członkiem prezydium ATypI i pełnomocnikiem ONZ ds. kultury rosyjskiej Maksimem Żukowem zafascynowała widzów aktualnymi pracami nad nowymi krojami alfabetów cyrylicy i języków arabskich.

Na walnym zgromadzeniu członków stowarzyszenie wybrało nowe prezydium. Prezydentem po raz kolejny wybrany został Mark Batty z amerykańskiej firmy International Typeface Corporation (ITC), w skład prezydium weszli także m.in. Maksim Żukow i Jean François Porchez.

Stowarzyszenie ATypI. Międzynarodowe Stowarzyszenie Typograficzne ATypI jest jedyną organizacją zrzeszającą osoby ze świata typografii, książki i druku o zasięgu ogólnoświatowym. Działa od 1957 roku i obecnie liczy ok. 400 członków.

ATypI działa na rzecz rozwoju i kultywowania typografii, zajmuje się problematyką ochrony prawnej krojów pism, ich katalogowaniem i klasyfikacją. Stowarzyszenie wydaje własne broszury informacyjne, książki i raporty. Coroczne konferencje, odbywające się w różnych miastach świata, służą jako miejsce spotkań, szkoleń i wymiany informacji. W roku 1975 konferencja ATypI odbyła się w Warszawie i Krakowie, ostatnio stowarzyszenie gościło swych członków w Parmie, Antwerpii, Budapeszcie, San Francisco, Barcelonie, Hadze, Reading w Wielkiej Brytanii i – w tym roku – w Lyon. Przyszłoroczna konferencja ma się odbyć w Bostonie, zaś konferencja milenijna – w Lipsku. ATypI finansowane jest ze składek członkowskich i z dotacji od sponsorów, głównie firm działających w przemyśle wydawniczym (Adobe, Agfa/Monotype, Font Bureau, International Typeface Corporation, Linotype Library, Microsoft).

Zmarły w 1992 roku znakomity polski wydawca i typograf, Roman Tomaszewski, intensywnie uczestniczył w pracach ATypI od 1974 roku, działając w licznych komitetach problemowych i będąc m.in. członkiem prezydium tej organizacji.

Więcej informacji na temat stowarzyszenia ATypI można znaleźć pod adresem: `http://www.atypi.org/`, gdzie zamieszczono również program konferencji. Obszerna relacja z MultiTypo ’98 znajduje się też pod adresem `http://www.fontzone.com/webtypo/`.

O formacie OpenType piszę w artykule „Sparing” zamieszczonym na stronie 49. FontLab: `http://www.fontlab.com/`. Kyrillitsa 99: `http://www.paratype.ru/news/cyr99.asp` i `http://www.paratype.ru/news/Cyr_res.html`.

◇ Adam Twardoch
twardoch@font.org