



# cmdfont zamiast instrukcji special

Bogusław Jackowski  
Krzysztof Leszczyński

BachTeX 2001

cmdfont.

Jak wygląda „special”?

ó.ñ.ř.č.ř.ó.ř.í.t

Jak wygląda „special”?



TEX

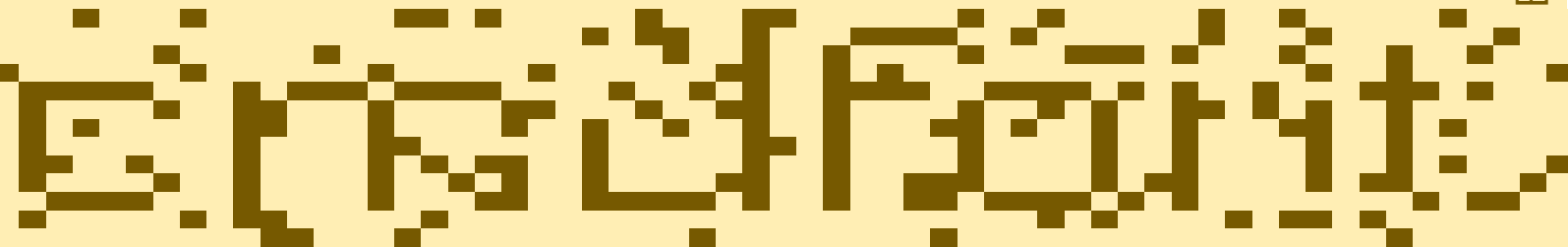
# Jak wygląda „special”?



TEX



METAFONT



# Jak wygląda „special”?



TEX



METAPOST METAFONT



# Czymże specjalnym jest METAPOST-owy special?

Wszystkie METAPOST-owe instrukcje `special` migrują na początek wynikowego pliku EPS; ich kolejność zostaje zachowana. T<sub>E</sub>X-owe instrukcje `\special`, które mogą się pojawić w przetwarzanym przez METAPOST-a kodzie T<sub>E</sub>X-owym, są po prostu ignorowane.

Przykład irytującego (dotychczas) problemu: jak pokolorować **wybrane fragmenty** tekstu T<sub>E</sub>X-owego?

Ogólniejszy problem: jak wykorzystać całe bogactwo istniejących pakietów T<sub>E</sub>X-owych, zwykle intensywnie korzystających z instrukcji `\special`?

# ROZWIĄZANIE

**Remedium jest użycie specjalnego fontu.**

Jeśli teksty, które miałyby być składane pewnym wybranym fontem, bylibyśmy w stanie *interpretować* (np. za pomocą zewnętrznego procesora) zamiast *składać*, moglibyśmy obejść problem ograniczeń narzuconych na METAPOST-owe instrukcje `special`.



## cmdfont: kto zacz?

```
designsize:=10bp/pt - epsilon;
fontdimen 2: designsize;
fontmaking:=1;

for i:=0 upto 255:
  beginfig(i-256);
    charwd:=charht:=chardp:=charic:=0;
  endfig;
endfor

end.
```

# Jak wygląda skład?

- Z użyciem operatora `infont`:

```
draw "META POST" infont "cmr10";
```

```
...  
(META POST) cmdfont 20 fshow  
...
```

- Z użyciem konstrukcji `btex ... etex`:

```
draw btex \font\f=cmdfont \f META POST etex;
```

```
...  
(META) cmdfont 10 fshow  
9.9999 0 moveto  
(POST) cmdfont 10 fshow  
...
```

# W jaki sposób wykorzystując cmdfont pokolorować materiał T<sub>E</sub>X-owy?

```
verbatimtex
\def\incmyk#1#2{%
  \leavevmode\rlap{\font\f=cmdfont \f
    gsave #1 setcmykcolor}%
  #2\hbox{\font\f=cmdfont \f grestore}}
etex
beginfig(100);
draw btex \vbox{
  \hsize 50mm \pretolerance10000
  \raggedright \noindent
  Prosty sposób pokolorowania \incmyk{1 0 1 0}
  {\bf wybranego fragmentu} tekstu.} etex;
endfig;
```

## Wynik kolorowania:

Prosty sposób pokolorowania  
wybranego fragmentu tekstu.

# To byłoby piękne, gdyby METAPOST był w stanie przetwarzać praktycznie dowolne pliki T<sub>E</sub>X-owe!

Daje się to osiągnąć poprzez przedefiniowanie instrukcji `\special` w taki sposób, by jej argument, umieszczony w pudełku (`\hbox`) o rozmiarach zerowych, był składany fontem `cmdfont`. Wynikowe pliki postscriptowe należy przetworzyć z użyciem narzędzi zewnętrznych tak, aby napisy składane fontem `cmdfont` zamienić na stosowne fragmenty kodu postscriptowego (tak jak by to uczynił sterownik `dvips`, natrafiwszy na odpowiednie instrukcje `special` w pliku DVI).

## Niektóre instrukcje `\special` są kłopotliwe dla METAPOST-a

Instrukcje `special` zawierające znacznik `,papersize'` najlepiej pominąć.

Instrukcje `special` zawierające znacznik `,ps::'` (proszę zwrócić uwagę na podwójny dwukropek) są nieco mętnie opisane (z grubsza chodzi o wstawianie kodu postscriptowego bez żadnych „przydatków”). Wróżą wpadki.

Instrukcje `special` zawierające znacznik `,!'` należy zebrać w postaci nagłówków postscriptowych – najprościej można to zrobić zapisując odtworzone instrukcje `\special` w pomocniczym pliku T<sub>E</sub>X-owym.

## **Bardzo ważne są instrukcje special zawierające znacznik ,psfile'**

Wstawiają je na przykład makra z pakietu epsf towarzyszącego sterownikowi dvips.

Argument takiej instrukcji zawiera parametry wyznaczające prostokąt ograniczający (*bounding box*), mianowicie `llx`, `lly`, `urx` i `ury`.

Kuszące jest skorzystanie w trakcie przetwarzania zewnętrznego z pozornie odpowiadających tym parametrom instrukcji postscriptowych `@llx`, `@lly`, `@urx` i `@ury`, z których korzysta także dvips.

Następującą instrukcję `\special`:

```
\special{PSfile=tiger.ps llx=22 lly=171  
        urx=567 ury=738 rwi=283 clip}
```

chciałoby się zamienić na:

```
@beginspecial  
  22 @llx 171 @lly 567 @urx 738 @ury  
  283 @rwi  
@clip  
@setspecial
```



... wynik wszakże można  
uznać za dyskusyjny:

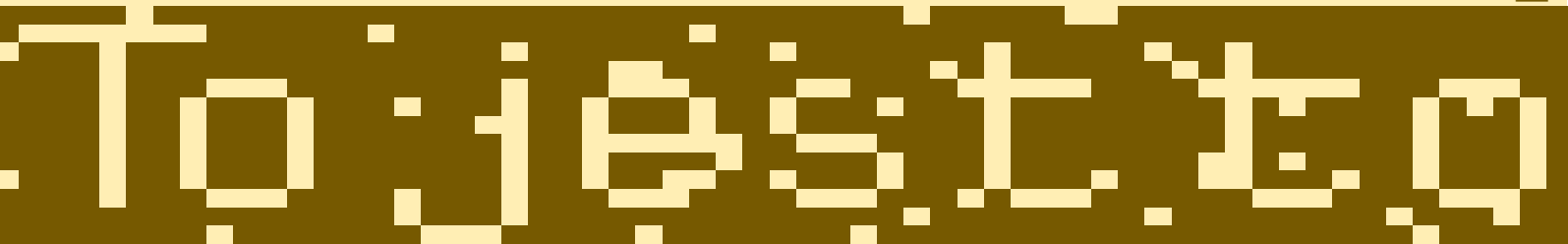


# Podsumowanie

Do czego może się przydać `cmdfont`?

Oto stan naszej wiedzy na wtorek,  
1 maja 2001, godz. 12.50:

- możemy wstawiać własne fragmenty kodu postscriptowego w wybranych miejscach w wynikowych plikach METAPOST-owych
- możemy rozszerzyć zestaw postscriptowych operatorów wykorzystywanych przez METAPOST-a (`eofill`, `eoclip`)
- możemy wzbogacić konstrukcję `btex ... etex` (np. o kolorowanie fragmentów tekstu)
- możemy wstawiać znaczniki w plikach  $\TeX$ -owych i METAPOST-owych do dalszego przetwarzania
- niemalże dowolny plik  $\TeX$ -owy może zostać użyty jako METAPOST-owy obiekt `picture`



Wykorzystanie fontu specjalnego  
otwiera nowe perspektywy przed  
zastosowaniami związanymi  
z instrukcjami `special`  
i przetwarzaniem zewnętrznym.  
Szczерze mówiąc, nie do końca  
jesteśmy świadomi nowych  
możliwości. Jednakże czujemy się  
w obowiązku przestrzec, że na tej  
jakże obiecującej drodze czyhać  
może wiele niebezpieczeństw  
w rodzaju rozbrykanych tygrysków...

To jest to

