



# Tworzenie dokumentów PDF w systemie T<sub>E</sub>X

Piotr Bolek

Strona główna

Strona tytułowa

Spis treści



Strona 1 z 13

Powrót

Pełny ekran

Zamknij

Koniec



Strona główna

Strona tytułowa

Spis treści



Strona 2 z 13

Powrót

Pełny ekran

Zamknij

Koniec

# Spis treści

<b>1</b>	<b>Wprowadzenie</b>	<b>3</b>
<b>2</b>	<b>Pakiety wspomagające tworzenie dokumentów PDF</b>	<b>6</b>
2.1	Pakiet hyperref . . . . .	6
2.2	Pakiet pdfscreen . . . . .	7
<b>3</b>	<b>Parametryzacja</b>	<b>8</b>
3.1	Grafika . . . . .	9
3.2	Wspomaganie przetwarzania . . . . .	12



# 1. Wprowadzenie

Efektywne wykorzystanie możliwości jakie dają współczesne formy prezentacji dokumentów i współczesne formaty prezentacyjne wymaga użycia odpowiednich narzędzi oraz zachowania pewnych konwencji przy przygotowywaniu tekstu dokumentu. Jedną z technologii umożliwiającą wykorzystanie możliwości przygotowania dokumentów i prezentacji ekranowych jest PDF. System  $\text{T}_{\text{E}}\text{X}$  znakomicie nadaje się do przygotowywania dokumentów w sposób elastyczny, tak aby przygotowywany dokument mógł być z jednego źródła formatowany na specyficzne dla różnych form prezentacji sposoby. W praktyce dokument zawsze powinien nadawać się do wydrukowania oraz coraz częściej do czytania z ekranu komputera. Obie formy prezentacji wymagają innego potraktowania: w wersji papierowej mamy dużą rozdzielczość, ale żadnych możliwości wspierania automatycznej nawigacji po dokumencie; w wersji ekranowej rozdzielczość jest stosunkowo niewielka, ale można używać odsyłaczy hipertekstowych, aktywnych indeksów, spisów treści, zakładek itd. Dodatkowo proporcje pola tekstu są inne. W praktyce także w dokumentach papierowych nie ma możliwości wykorzystania koloru, w dokumentach ekranowych wykorzystanie kolorów może podnieść atrakcyjność i ułatwić czytelnikowi sprawne przyswajanie treści.

Przygotowanie dokumentów pozwalających na otrzymanie dwóch wersji: ekranowej (przeznaczonej do czytania z ekranu) i drukowalnej (przeznaczonej do wydrukowania) nie jest trudne, chociaż wymaga pewnej dyscypliny przy tworzeniu dokumentu.

Możliwe sposoby przetwarzania dokumentów pokazane są na rysunku 1

Ostateczna postać dokumentu pokazana na rysunku to 3 (Postscript) lub 4 (PDF), co w naszym przypadku oznacza dokument drukowalny oraz ekranowy. Jak widać do wersji ekranowej można dojść różnymi ścieżkami: bezpośrednio –

Strona główna

Strona tytułowa

Spis treści



Strona 3 z 13

Powrót

Pełny ekran

Zamknij

Koniec



Strona główna

Strona tytułowa

Spis treści



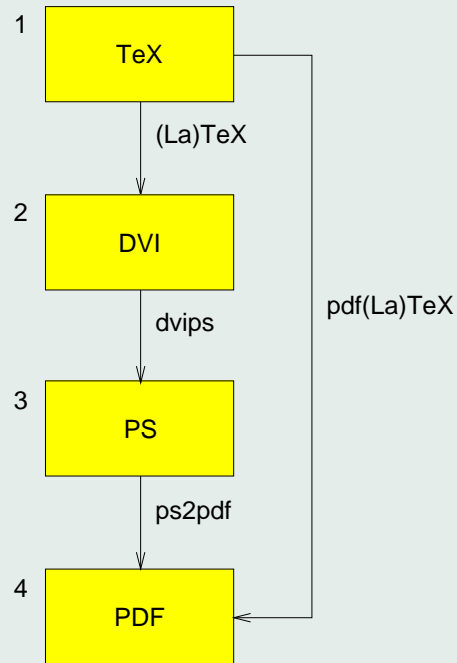
Strona 4 z 13

Powrót

Pełny ekran

Zamknij

Koniec



Rysunek 1: Przetwarzanie dokumentu



wykorzystując program pdfTeX, albo ścieżką „tradycyjną” wykorzystując niezmodyfikowaną wersję programu tex, sterownik postscriptowy (dvips) oraz program „destylujący” (np. ps2pdf z pakietu ghostscript).

Strona główna

Strona tytułowa

Spis treści



Strona 5 z 13

Powrót

Pełny ekran

Zamknij

Koniec



## 2. Pakiety wspomagające tworzenie dokumentów PDF

Do efektywnego wykorzystania możliwości jakie daje format PDF nie jest potrzebna znajomość szczegółów technicznych związanych z tym formatem. Pakiety `hyperref` i `pdfscreen`, umożliwiają wykorzystanie możliwości formatu PDF nawet niezbyt zaawansowanym użytkownikom  $\text{\LaTeX}$ -a.

### 2.1. Pakiet `hyperref`

Podstawowym celem pakietu `hyperref` jest dodanie cechy interaktywności do standardowych mechanizmów odwołań. Dzięki temu konstrukcje takie jak spisy treści, odwołania do rysunków, tabel, wzorów, pozycji bibliograficznych oraz innych części dokumentu stają się aktywnymi odsyłaczami hipertekstowymi. Utworzenie interaktywnej wersji dokumentu nie wymaga jakiegokolwiek modyfikacji kodu źródła dokumentu poza włączeniem w nagłówku pakietu `hyperref`.

Dodatkowo pakiet `hyperref` umożliwia jawne definiowanie odsyłaczy hipertekstowych, oraz daje dostęp do standardowych funkcji przeglądarki Acrobat Reader takich jak: zamknięcie dokumentu, zmiana powiększenia wyświetlania dokumentu, drukowanie, wyszukiwanie tekstu, nawigację po dokumencie, itd. Użycie tych dodatkowych możliwości sprawia, że do przetwarzania dokumentu musimy używać odpowiedniej wersji programu  $\text{\TeX}$  oraz pakietu `hyperref`, co utrudnia tworzenie dokumentów umożliwiających przygotowywanie dokumentów w taki sposób żeby możliwe było z jednych źródeł wygenerowanie dokumentu drukowalnego oraz ekranowego. Odpowiednia **parametryzacja** dokumentu opisana dalej pozwala uniknąć tego typu problemów.

Pakiet `hyperref` udostępnia wiele opcji wywołania określających różne pa-

Strona główna

Strona tytułowa

Spis treści



Strona 6 z 13

Powrót

Pełny ekran

Zamknij

Koniec



Strona główna

Strona tytułowa

Spis treści



Strona 7 z 13

Powrót

Pełny ekran

Zamknij

Koniec

ramtry związane np. ze sposobem prezentacji odsyłaczy hipertekstowych. Szczegółowy opis dostępnych opcji i ich znaczenie można znaleźć w dokumentacji od pakietu.

## 2.2. Pakiet pdfscreen

Pakiet `hyperref` umożliwia łatwe przekształcenie zwykłego dokumentu przygotowanego pierwotnie do tradycyjnej formy prezentacji w postaci papierowej na dokument elektroniczny wykorzystujący możliwości formatu PDF. Jednak w praktyce, ze względu na bardzo istotne różnice dostępnej na ekranie i w druku rozdzielczości oraz inne proporcje typowej kartki papieru i ekranu komputera dokument przygotowany do prezentacji klasycznej będzie kłopotliwy przy czytaniu z ekranu komputera.

Rozwiązaniu tego problemu służy pakiet `pdfscreen`. Pakiet ten umożliwia łatwe tworzenie wygodnych dla czytelnika dokumentów ekranowych. Oprócz zapewnienia ustawienia odpowiednich dla prezentacji ekranowej proporcji i wymiarów pola prezentacji tekstu, pakiet ten umożliwia tworzenie interaktywnego panelu ułatwiającego nawigację po dokumencie. Możliwe jest skorzystanie z panelu predefiniowanego, albo zdefiniowanie go samodzielnie od początku.



Strona główna

Strona tytułowa

Spis treści



Strona 8 z 13

Powrót

Pełny ekran

Zamknij

Koniec

### 3. Parametryzacja

Dla ułatwienia pracy z dokumentem, który ma być udostępniany na dwa sposoby najlepiej zastosować automatyczne wykrywanie wykorzystywanej wersji T<sub>E</sub>X-a i w razie użycia do formatowania programu pdftex stosować konwencje dla dokumentu ekranowego, a po użyciu tex- a klasycznego tworzyć wersję drukowalną.

Pierwszym i niezbędnym warunkiem pozwalającym na sparаметryzowanie dokumentu jest wykrycie jakiego używamy programu formatującego. Jednym ze sposobów może być sprawdzenie czy są zdefiniowane polecenia dostępne wyłącznie w programie pdftex. Takim poleceniem jest `\pdfoutput`. Dodatkowo warto zdefiniować nowe polecenie warunkowe pozwalające zapamiętać jaki rodzaj dokumentu tworzymy. Poniższy fragment kodu, który najlepiej zapisać jako oddzielny pakiet (np. `ifpdf.sty`) definiuje nowe polecenie warunkowe (`\ifpdf`) i w zależności od użytej wersji programu tex ustawia wartość zwracaną przez to polecenie.

```
\newif\ifpdf\pdftrue
\ifx\pdfoutput\undefined\pdffalse\fi
```

Po wstawieniu takiego kodu do dokumentu można łatwo parametryzować dokument np.:

```
\ifpdf
  \usepackage[screen,panelleft]{pdfscreen}
\else
  \newcommand\hypertarget[2]{#2}
  \newcommand\hyperlink[2]{#2}
\fi
```





Strona główna

Strona tytułowa

Spis treści



Strona 9 z 13

Powrót

Pełny ekran

Zamknij

Koniec

W powyższym przykładzie dla sformatowania wersji ekranowej używamy pakietu `screen`, a w wersji drukowalnej redefiniujemy niedostępne polecenia.

Kod sprawdzający, jakiej wersji programu używamy najlepiej umieścić w oddzielnym pakiecie (pliku z rozszerzeniem `.sty`) i włączać w nagłówku dokumentu poleceniem `\usepackage`:

```
\usepackage{ifpdf}
```

### 3.1. Grafika

W dokumentach texowych można wstawiać grafikę. Obie wersje programu – klasyczna i `pdftex` potrafią dołączać do dokumentu pliki graficzne w różnych formatach graficznych:

**L<sup>A</sup>T<sub>E</sub>X:** w zasadzie wyłącznie EPS, w tym także MPS (efekt działania programu Metapost), będący specyficzną odmianą formatu EPS. Inne rodzaje plików graficznych (np. `wmf` w implementacjach dla systemów MS), nawet jeśli są obsługiwane to prawie na pewno spowodują problemy z przenośnością.

**pdftex:** PDF, MPS, PNG, JPG, TIF.

W przypadku klasycznego L<sup>A</sup>T<sub>E</sub>X-a najlepszym rozwiązaniem jest używanie *zawsze* formatu EPS. Najlepszym sposobem na rozwiązanie problemu grafiki jest pokazana automatyczna konwersja do odpowiedniego formatu z wykorzystaniem możliwości programu `make` (patrz p. 3.2).

Wygodne dołączanie rysunków do dokumentu umożliwia pakiet `graphicx`. Udostępnia on polecenie `\includegraphics`, którego obowiązkowym argumentem jest nazwa pliku (prawie zawsze powinna być podana bez rozszerzenia – do wyjątków należą pliki wygenerowane przez Metapost). Nie podając rozszerzenia



Strona główna

Strona tytułowa

Spis treści



Strona 10 z 13

Powrót

Pełny ekran

Zamknij

Koniec



Rysunek 2: Mapa bitowa

włączanych plików graficznych umożliwiamy wykorzystywanie zapisanych w pakiecie `graphicx` reguł umożliwiających automatyczne włączanie odpowiedniego pliku graficznego rozpoznawanego po rozszerzeniu. Wystarczy wtedy przygotować dwie wersje tego włączanej grafiki w formatach akceptowanych przez obie wersje  $\text{T}_{\text{E}}\text{X}$ -a,

Rysunek 2 był włączony w następujący sposób:

```
\begin{figure}[htb]
  \begin{center}
    \includegraphics[width=0.4\hsize]{motyl}
```



```
\caption{Mapa bitowa}
\label{fig:bitmapa}
\end{center}
\end{figure}
```

Strona główna

Strona tytułowa

Spis treści



Strona 11 z 13

Powrót

Pełny ekran

Zamknij

Koniec

Dostępność narzędzi do wsadowej konwersji plików graficznych sprawia, że w praktyce nie ma potrzeby przechowywania obrazków w więcej niż jednym formacie: dla plików z grafiką obwiedniową powinien być to format EPS, dla plików z grafiką rastrową powinien być to format JPG, PNG lub TIF.

Do konwersji EPS  $\rightarrow$  PDF należy użyć dostępnego we wszystkich współczesnych dystrybucjach  $\text{\TeX}$ skryptu `epstopdf`. Wykorzystuje on do konwersji program `ghostscript` (dla uzyskania dobrej jakości fontów konieczne jest używanie `ghostscripta` w wersji  $\geq 6.0$ ).

Do konwersji TIF/JPG/PNG  $\rightarrow$  EPS można użyć dowolnego programu konwertującego np. z pakietów `netpbm`, `pbmplus` albo `ImageMagick`, osobiście najbardziej polecam ten ostatni pakiet.

Poniższa tabelka zawiera przepisy, na niezbędną konwersję w zależności od pierwotnego formatu, w jakim jest zapisany plik graficzny:

Format	$\text{\LaTeX}$	pdf $\text{\LaTeX}$
EPS	bez konwersji	<code>epstopdf</code>
TIF	<code>ImageMagick</code>	bez konwersji
PNG	<code>ImageMagick</code>	bez konwersji
JPG	<code>ImageMagick</code>	bez konwersji
MPS	bez konwersji	bez konwersji



Strona główna

Strona tytułowa

Spis treści



Strona 12 z 13

Powrót

Pełny ekran

Zamknij

Koniec

## 3.2. Wspomaganie przetwarzania

W celu zautomatyzowania przetwarzania najlepiej skorzystać z możliwości programu `make`. Przykładowy uniwersalny plik `Makefile` umożliwia automatyczne przetwarzanie dokumentów z uwzględnieniem zależności między plikami. Pokazany poniżej plik `Makefile` (używany przy przetwarzaniu tego dokumentu) jest przygotowany w taki sposób, że wystarczy zmieniać tylko pierwsze cztery wiersze.

- Pierwszy wiersz zawiera nazwę głównego pliku dokumentu.
- Drugi zawiera nazwy plików, od których zależy dokument główny. Mogą to być tak jak tutaj pliki ze stylami, albo zapisane w oddzielnych plikach części dokumentu włączane poleceniami `\input` albo `\include`.
- Kolejny wiersz zawiera nazwy plików EPS, które należy utworzyć przez konwersję z formatu bitmapowego, aby możliwe było wygenerowanie wersji drukowalnej dokumentu.
- Ostatni wiersz zawiera nazwy plików, które należy utworzyć przez konwersję EPS → PDF. Pierwotnym formatem tych plików jest EPS, nadają się więc do umieszczenia w wersji drukowalnej dokumentu, a do wersji ekranowej muszą być przekonwertowane programem `epstopdf`

Wszystkie zależności i reguły konieczne do automatycznego przetworzenia plików graficznych oraz dokumentów są zdefiniowane w dalszej części pliku `Makefile`.

```
TEX= pdf.tex
INCLUDES= ifpdf.sty mypanel.sty
PAPERFIG= motyl.eps
SCREENFIG= proces.pdf
```

```
##### Zmieniać tylko do tego miejsca
```

```
PDFLATEX = pdflatex
LATEX    = latex
```



Strona główna

Strona tytułowa

Spis treści



Strona 13 z 13

Powrót

Pełny ekran

Zamknij

Koniec

```
DVI := $(patsubst %.tex,%.dvi,$(TEX))
PS := $(patsubst %.tex,%.ps,$(TEX))
PDF := $(patsubst %.tex,%.pdf,$(TEX))
```

```
all: ps pdf
ps: $(PS)
pdf: $(PDF)
```

```
$(DVI): $(PAPERFIG) $(INCLUDES)
$(PDF): $(SCREENFIG) $(INCLUDES)
```

```
EXTRACLEAN=$(PAPERFIG) $(SCREENFIG)
```

```
%.dvi: %.tex
    rm -f $(basename $<).aux
    rm -f $(basename $<).toc
    $(LATEX) $<
    $(LATEX) $<
    $(LATEX) $<
```

```
%.pdf: %.tex
    rm -f $(basename $<).aux
```

```
rm -f $(basename $<).toc
$(PDFLATEX) $<
$(PDFLATEX) $<
$(PDFLATEX) $<
```

```
%.pdf: %.eps
    epstopdf $<
%.eps: %.jpg
    convert $< $@
%.eps: %.png
    convert $< $@
%.eps: %.tif
    convert $< $@
```

```
%.ps: %.dvi
    dvips $<
```

```
clean:
    rm -f $(DVI) $(PS) $(PDF)
    rm -f *.log *.aux *.out *.toc
    rm -f $(EXTRACLEAN)
```

Polecenia uruchamiające przetwarzanie dokumentu są następujące: **make pdf** aby otrzymać wersję ekranową, albo: **make ps** aby otrzymać wersję drukowalną. Podanie polecenia: **make** bez żadnych dodatkowych argumentów jest równoważne podaniu polecenia: **make ps pdf** czyli spowoduje wygenerowanie dokumentu w wersji drukowalnej i ekranowej.