

本稿は [Linux Japan 誌](#) 2001 年 10 月号に掲載された記事に補筆修正したものです。

PDF とハイパーテキスト

Linux Japan の誌面が 3 段で少し密度の高いレイアウトに変わりました。筆者もそれに応じて、内容の濃い記事をとめるのですが、今までの Linux との付き合い方を変えることもできませんので、ちょっとした話題を中心に徒然に語るというスタイルを続けていこうと思っています。さて今回は、数式を含む文書の作成とネットワーク配信というお題で [dvipdfm](#)^[1] と [hyperref.sty](#)^[2] を取り上げます。お付き合いの程、どうぞ宜しくお願い致します。

我々理工学系の人間にとって数式はなくてはならない重要な要素です。WEB で講義教材を公開するには、数式を画像に変換して埋め込んだ HTML を書く ([latexhtml](#)^[3]) か、文書を PostScript 化するかでした。ところが、ネットワーク時代の電子文書ファイルに進化した PostScript という性格の PDF (Portable Document File) のプレビュー Acrobat Reader が Adobe 社から配布されるようになって、事態は変わりました。PS と同様にプラットフォームを選ばずページレイアウトを維持しつつも、軽く可搬性の高い PDF が少なくとも簡単に見ることができるようになったからです。ところで、PDF を作成するにはどうしていたのでしょうか？ 実は Ghostscript が PDF への出力をサポートしていて [ps2pdf](#) という (実体は `gs` を呼び出すシェルスクリプト) 変換ツールのお世話になっていました。しかし、この [ps2pdf](#) は日本語フォントをビットマップで埋め込んでしまい、サイズが大きくなるという欠点がありました。せっかくの PDF の軽さが活かないままだったのです。さらにビットマップで埋め込まれた日本語フォントは、あまり綺麗に表示されません (とくに Acrobat では汚いです)。MS-Windows の Acrobat を用いればいいのですが、筆者は原理主義なので Linux だけで何とかならないかと思っていましたところ、 $\text{T}_{\text{E}}\text{X}$ のユーザーに朗報がもたらされました。dvipdfm という DVI から PDF への変換ツールの日本語化パッチを平田さんが作成され、十分実用になっているのです^[4]。図 1 は、[ps2pdf](#) と [dvipdfm](#) で得られた PDF の日本語フォントの表示結果です。一目瞭然で [dvipdfm](#) に軍配があがります、サイズも [ps2pdf](#) の場合 ~240k、[dvipdfm](#) の場合 ~100k となり、こちらも [dvipdfm](#) の圧勝です。というわけで、これからは [dvipdfm](#) しかないという結論です。

PDF ファイルの作り方

従来の [ps2pdf](#) とこれから紹介する [dvipdfm](#) の比較について、[土村さん](#)^[5] が要領良くまとめていらっしゃいますので、一読を薦めます。普段 $\text{T}_{\text{E}}\text{X}$ を使わない方のために、 $\text{T}_{\text{E}}\text{X}$ で文書を書き印刷するまでの工程を簡単に記します。図 2 が概略です。ソースファイル (拡張子 `.tex`) を `emacs` などのエディタで編集し、`latex` などでタイプセットし DVI ファイル (拡張子 `.dvi`) に変換します。DVI は様々なビューアがありますが、現在は `xdvi(k)` が標準的にインストールされているようです。さらに `dvips(k)` で PostScript ファイル (拡張子 `.ps`) に変換し、Ghostscript で X11 に表示させたり、各種プリンタに印刷を実行させます。

今までは、PS で終わっていましたが、冒頭述べましたようにさらに PDF ファイルの需要が高まってきており、従来は [ps2pdf](#) (実体は Ghostscript のシェルスクリプト) で PS → PDF と変換していました。dvipdfm は途中の PS への変換を飛び越して DVI → PDF と、一気に変換するツールです。

さて $\text{T}_{\text{E}}\text{X}$ のソースから直接 PDF を生成するツールが欲しいとなるでしょう。もちろん [pdftex](#)^[8] があります。MS-Windows では日本語が通るものが既にあるみたいです ($\text{T}_{\text{E}}\text{X}$ の日本語化に関しては MS-Windows 先行していて充実してますね)。

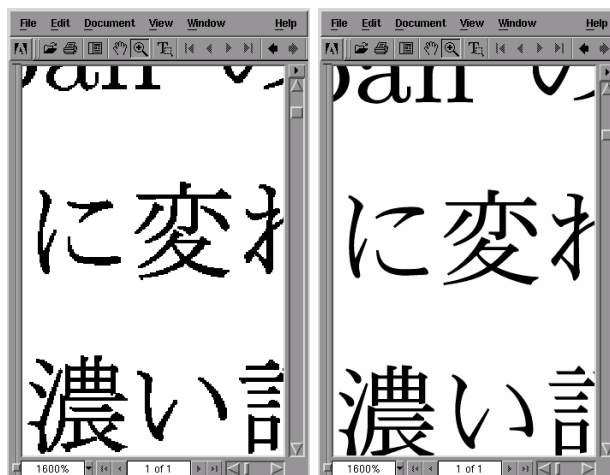


図 1 Ghostscript と [dvipdfm](#) で変換した PDF の日本語フォントの比較: [dvipdfm](#) (右) はスケラブルのままなので綺麗に表示される。Font が異なっているがどちらも元はスケラブル。

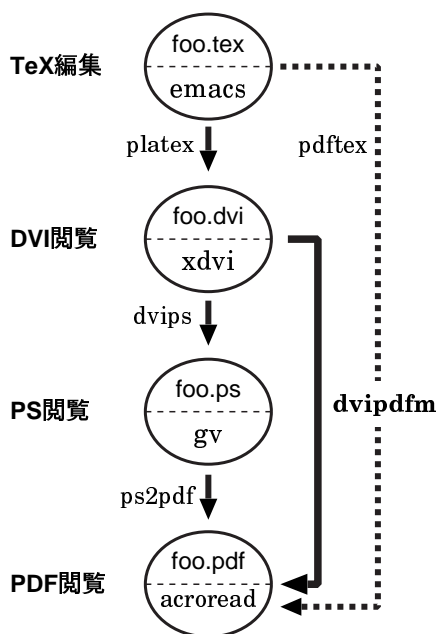


図 2 TeX のソースから PDF を作成する工程の概略

dvipdfm

ハイパーテキストやノートやしおりなど、インターネット時代の電子文書に特化した PDF を TeX の DVI から直接作成するツールが dvipdfm です。基本的に自分でソースからコンパイルしてインストールしなければなりません。バイナリ配布をお望みならば、RPM パッケージを土村さん [5][W3] が作成しておられますし、Plamo 用のパッケージは筆者のホームページ [6][W3] にあります。

kpathsea ライブラリ

ソースからインストールする場合に少し面倒なことがあります。kpathsea のヘッダーが必要 (/usr/include/kpathsea) なのです。ライブラリ自体はインストールされていても、ヘッダーまでは入っていない場合がありますから、まずこれをインストールしなければなりません。するとここで戸惑うことがあります。というのは kpathsea 単独では配布されていないのです。そこで xdvik, dvipsk, dviIjk など kpathsea を含むアプリケーションのソースを利用することになります。以下ではファイルサイズの最も小さい dviIjk を例に方法を説明します (xdvik でも dvipsk でも同様です)。まずは、恒例の CTAN [7][W3] の dviware から dviIjk.tar.gz を入手してください。最初にソースを展開

して configure を行い、kpathsea ディレクトリに移って make を実行します。すると c-auto.h と paths.h というホスト固有の TeX のディレクトリ構成を反映したヘッダーファイルが出来上がります。これらを含めて、全てのヘッダーファイルを /usr/include/kpathsea ディレクトリにコピーします。具体例は以下のようになります。

```
# tar zxvf dviIjk.tar.gz
# cd dviIjk-***
# ./configure
# cd kpathsea
# make
# mkdir /usr/include/kpathsea
# cp *.h /usr/include/kpathsea
```

dvipdfm のコンパイル

以上で kpathsea のヘッダーファイルが見えるようになりましたから、dvipdfm のコンパイルが素直に行えます。ソースを展開して、日本語化パッチをあて、texmf がどこにあるかに合わせて configure を行い make です。具体的には

```
# tar zxvf dvipdfm-0.13.2c.tar.gz
# cd dvipdfm-0.13.2c
# patch -p1 < dvipdfm-jpatch-p0d.patch
# ./configure --prefix=/usr
# make
# make install
# mktexlsr
```

最後の mktexlsr を忘れないでください。

PDF への変換

dvips と似ていて以下のように実行すると foo.pdf が作成されます。-p a4 は用紙の指定オプションです。オプション --help で概略が判ります。また設定ファイルは \$TEXMF/dvipdfm/config/config です。

```
dvipdfm -p a4 foo.dvi
```

hyperref.sty

インストール

hyperref.sty は標準ではインストールされていないと思いますので、CTAN の macros/latex/contrib/supported/hyperref 以下のファイル入手して、\$TEXMF/tex/latex/hyperref などに入れてください。また最近のもの

は `url.sty` を要求するらしいので、これも `macros/latex/contrib/others/misc/` から持ってきて、`$TEXMF/tex/latex/misc` などに入れてください。Kpathsea を利用している場合、TeX 関連のファイルを更新したら、必ず `mktexlsr` を実行します。検索パスに登録されたかどうかは以下のように `kpsewhich` で確認してください。

```
$ kpsewhich hyperref.sty
.../tex/latex/hyperref/hyperref.sty
```

`hyperref` のディレクトリに移動して `make` しますと、様々な `*.sty` や `*.def` ファイルができます。もし `latex` が無いというエラーが出たならば、`Makefile` 中の `latex` を `platex` に置き換えましょう。`hyperref.dvi` を作成しますが、最後に `makeindex` (索引の作成) で失敗するかもしれません。それでも一応索引なしの `hyperref.dvi` ができあがりますから、まあよしとしましょう。そうして、`mktexlsr` でインストール終了です。

マクロの使い方

`hyperref.sty` は元来 TeX が有していたハイパー文書 (図や式の参照、目次や参考文献の自動作成) の機能を、ネットワーク越しの HTTP に広げたものです。また、PDF 独特のコマンドを発効するマクロを取り入れているようですが、筆者は PDF に詳しくないので説明できません。平田さんのページにも PDF の多機能性を示すサンプルがありますし、もっと驚くような例を紹介しているページもありますので、一度覗いてみると良いでしょう [9][W3]。

ここではリンクに関するマクロだけを説明します。まず、目次や図や式や参考文献へのリンクには色がつき、クリックするとリンク先に飛びます (図 3)。文書の内部にタグを置くマクロは

```
\hypertarget{タグ名}{表示文字列}
```

です。この部分を参照してリンクをはるには

```
\hyperlink{タグ名}{表示文字列}
```

と記述します。文書の外部の URL へリンクをはるのは

```
\href{URL}{表示文字列}
```

です。ここでちょっと注意があります。URL で用いる文字でチルダ“~”は TeX ではエスケープして“\~”と記述しなければなりません (注意: 最近の `hyperref.sty` ではこのようにエスケープする必要がなくなりました)。外部 URL へのリンク箇所をクリックすると Acrobat

Reader では図 4 のようにブラウザの起動ダイアログが現れますから、Launch します。また、起動するブラウザは File Preferences Weblink で現れるダイアログで設定します (図 5)。

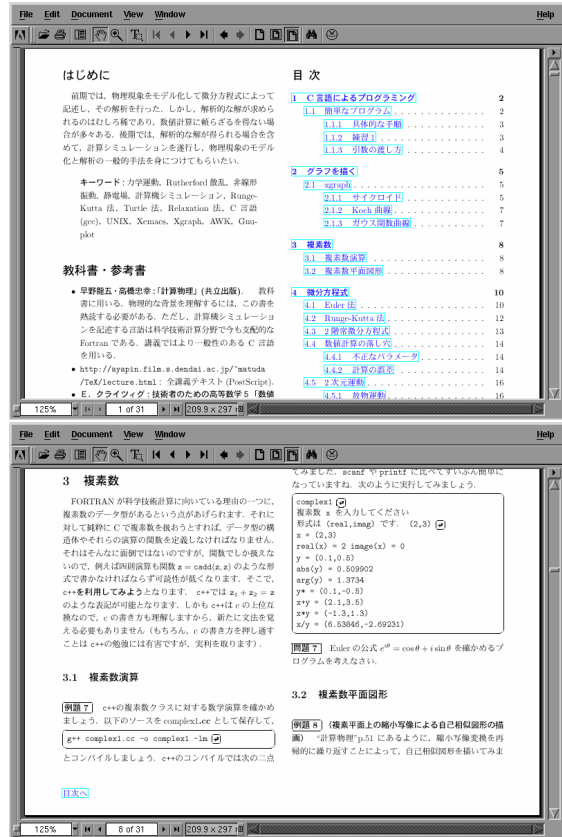


図 3 自動的に生成される目次もハイパーリンクになっており (上)、文書内の該当ページにクリック一発で飛んでいく。また脚注に、目次へのアンカーを埋めこむことも簡単にできる (下)。

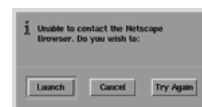


図 4 ブラウザーとして Netscape を起動するかを尋ねるダイアログ。

List 1 に簡単な TeX ソースを示しますので実際に PDF を作成してみてください。

```
$ platex hyperlink.tex
$ dvipdfm -p a4 hyperlink.dvi
$ acroread hyperlink.pdf
```

最小限の機能を確認するだけですが、エッセンスはこれだけと言っても過言ではないと思います。

List 1 hyperlink.tex



図 5 File Preferences Weblink で現れる，ブラウザの設定ダイアログ。

```

\documentclass[a4j]{jarticle}
\usepackage[colorlinks]{hyperref}

\begin{document}
\section{はじめに}
\hypertarget{はじめに}{}
\hyperlink{続いて}{続いてのページ}に進みます。

URL を理解するか
\href{http://ayapin.film.s.dendai.ac.jp/~matuda/Gnuplot}
{グラフは Gnuplot におまかせ}
をクリックしてください。

\newpage
\section{続いて}
\hypertarget{続いて}{}

\hyperlink{はじめに}{はじめにのページ}に戻りましょ
う。
\end{document}

```

次回は

これで数式を含んだハイパーテキストが簡単に作成できるようになりました。こうなると内容の充実に集中できます。ということでまたもや物理と数学のための数式処理ツールをおさらいしたいとおもいます。まずは MathMatica に劣らぬ完成度を有する Maxima です。このツールは Lisp(gcl) で書かれているので、Lisp を知らない筆者には全くの Black Box ですが、表面的な使い方の紹介だけでもかなり面白いのでは思っております。

参考文献

- [1] dvipdfm の開発者サイト
<http://gaspra.kettering.edu/dvipdfm/>
- [2] CTAN の下の次のディレクトリにあります。
<http://www.ring.gr.jp/pub/text/CTAN/macros/latex/contrib/supported/hyperref/>
- [3] 2000 年 2,3 月号の本連載記事。武藤さんの日本語化パッチ
<http://www.topstudio.co.jp/~kmuto/software/latex2html>
- [4] dvipdfm 日本語化のページ
<http://hamilcar.phys.kyushu-u.ac.jp/~hirata/dvipdfm/index.html.ja>
- [5] dvipdfm 日本語化 RPM パッケージ
<http://www.nn.iij4u.or.jp/~tutimura/tex/dvipdfm.html>
- [6] Plamo2.1 用
<http://ayapin.film.s.dendai.ac.jp/~matuda/Plamo21/plamo21.html>
- [7] Ring Sever を使いましょう
<ftp://www.ring.gr.jp/pub/text/CTAN/dviware/>
- [8] TUG (TeX Users Group) の PDFTeX 情報
<http://www.tug.org/applications/pdfTeX/>
- [9] PDF の記述能力を示す例題集 (リンク切れ?)
<http://www.pragma-ade.com/examples.htm>