

1 縦組み対応

1.1 TeX と PDF

日本語 TeX (および DVI フォーマット) では「組み方向」(ディレクション)と「フォント」は独立に扱われる。縦ディレクション中に横組み用フォントが使われることもあるし、横ディレクション中に縦組み用フォントが使われることもある。縦ディレクション中に横組み用フォント (欧文など) が使われると文字は 90 度時計周りに回転されなければならない。

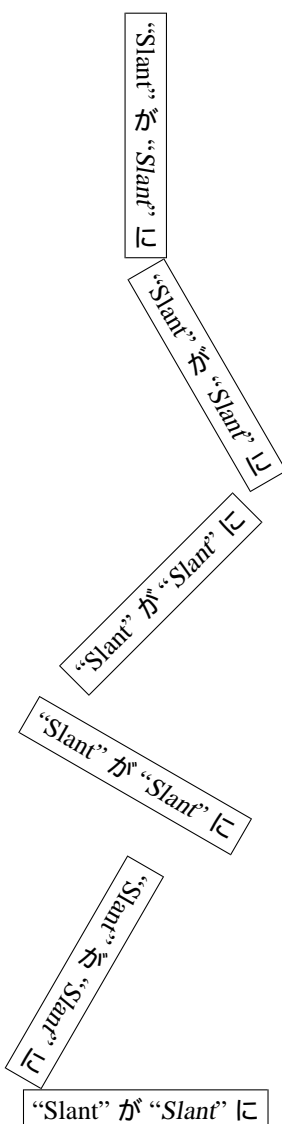
これに対して PDF と PostScript では、大雑把にいうと「横組み用フォントを使った横組み」と「縦組み用フォントを使った縦組み」があるだけである。PostScript や PDF で出力する DVI ドライバは、縦ディレクション中に横組み用フォントに代わったら文字がすべて回転されるようにする。Adobe-Japan1-3 ではあらかじめ回転されたアルファベットや数字が定義されているようだが、TeX ユーザがこれらを使いたいと思いつくはずないであろう。そもそも TeX ではこのようなことはやらない。

縦組み中に `\hbox{\yoko ...}` で横にしてみよう。この場合は横ディレクションで横組み用フォントが使われることになり、dvipdfm により生成された PDF では横組み用のエンコーディング「H」の Type 0 フォントが使用される。

1.2 斜体

dvipdfm でもスラップファイルに「-s 0.167」のような記述を書くことで *slanted* にすることができる。これは *text matrix* を調整して文字を变形する手段で実現されている。¹

ただし *slanted* を試してみよう。この文書では TeX フォントが使用されており、斜体としてデザインされたものは含まれない。 `\slshape` を使うと「-s 0.167」で変形された文字が描画されるようにしている。



¹ - /FontMatrix を使ったほうがよいかもしれないが、この方法は PDF 文書中に埋め込まれる Type 1 フォントと Type 3 フォントでは使えない。

なお、縦組み用和文フォントでは‘s’は無視するようにしている。対応するにしても、横に傾けるのではなく、縦方向に剪断変形をくわえるかたになるだろう。また‘extend’‘e’は縦組み用フォントでも『横』に広がるようになっている。

1.3 graphics パッケージとの互換性

graphics パッケージで利用可能ないくつかの機能は縦ディレクション中では正しく機能しない。

- `\scalebox` と `\updown` の機能がない。

例へて `\fbox{\scalebox{2}[1]{scale}}` と

`\scale`

を用いた場合、互換性も方向が間違っている。

- `\reflectbox` と `\updown` の機能がない。

`\fbox{\reflectbox{\reflect}}` と

`\reflect`

を用いて、互換性も軸が違っている。

- `\updown` と `\resizebox` の

`\fbox{\resizebox{1in}{2\height}{\resize}}` と次のように。

`\resize`

これらは `\updown` の機能として、これは箱のなかに文字がきれいに収まるはずである。

`\updown` の機能がない場合は `\graphics` パッケージの `dvipdfm.def` や `dvips.def` に記述されている `gsave` のマクロに関係している。

```
\def\Gscale@start{\special{ps: gsave currentpoint currentpoint translate
\Gscale@x\space\Gscale@y\space scale neg exch neg exch translate}}
```

DVI フォントに PostScript テキストの書き込みを可能にするため、PostScript に出力するためのドライバはこれらの内容をそのまま書き出すだけで済むが、そのためには `\PostScript` テキストを解釈し適切な形で処理しなければならない。

アスキー PEX 形式の `dvips` とは縦組み用の `dvips` とは `\scalebox` などの `\updown` の効果をもたらす。 `dvips` とは縦組みの場合には 0 度回転された座標系を基準としているからである。日本語 `dvipdfm` と `dvips` は異なる。一般に PostScript オペレータ `scale` を用いる

は縦スクリーンでは予期したように動作しない。

graphics パッケージを修正するに付添付通りの動作をやるに付する。 dvipdfm.def に \Gscale@start の定義を

```
\def\Gscale@start{\special{ps: gsave currentpoint currentpoint translate
\Gscale@x\space\Gscale@y\space\iftdir exch \fi scale neg exch neg exch translate}}
```

に変更するつもりだ。新たに加わるものは行間の \iftdir exch \fi による処理だ。これは縦スクリーンのときだけ x 方向 y 方向の伸長率を入れ換える働きをする PostScript コードを挿入するだけだ。このようにするものは DVI シンバ、特に dvips や xdvi は必ず対応しなくてはならない。

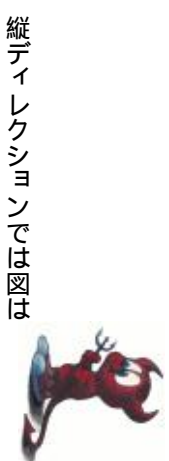
dvipdfm 専用になるものは今の Gscale@start に \Gscale@end の定義を置き換えてしまったほうが良いだ。

```
\def\Gscale@start{\special{pdf: bt
\iftdir
xscale \Gscale@y\space yscale \Gscale@x
\else
xscale \Gscale@x\space yscale \Gscale@y
\fi}}
\def\Gscale@end{\special{pdf: et}}
```

このようにするものは DVI シンバには無視されてしまう、全く機能しなくなる。

これらの処理を行なうに graphics パッケージが \usepackage[dvipdfm]{graphicx} のように dvipdfm オプションつきで使用されないと意味がない。また \iftdir は PostScript 特有のものであるにも注意しなければならない。単純に dvipdfm 側で縦スクリーンのときは PostScript オペレータ 'scale' の解釈をかえるようにするだけでもよいかもしれないが、ほかのところで影響がでる可能性もある。欧文専用 TeX の互換性も考えるに dvips と同じ態度をするのが正しいのかもわからないが、しばらく様子を見るつもりである。

1.4 縦組みと図



2 のように90度回転されて表示される。この例では

```
\special{pdf: image @beastie (daemon.jpg)}
```

としたが、\includegraphics{...}でも90度回転される。

中途半端に回転したり、縮小したりしてみよう。



これは

```
\includegraphics[scale=0.8, angle=-45]{daemon.jpg}
```

である。

縦組み中で\hbox{\yoko ...}と組み合わせると



のようになる。基本的には外部から取り込まれるPostScriptやJPEGなどの画像

は欧文扱いである。画像は描画されるときに必要に応じて回転される。あらかじめ回転された画像が取り込まれるのではない。

usexobj (uxobj) の場合は回転しないように扱う。すなわち

```
\special{pdf: uxobj @beastie}
```

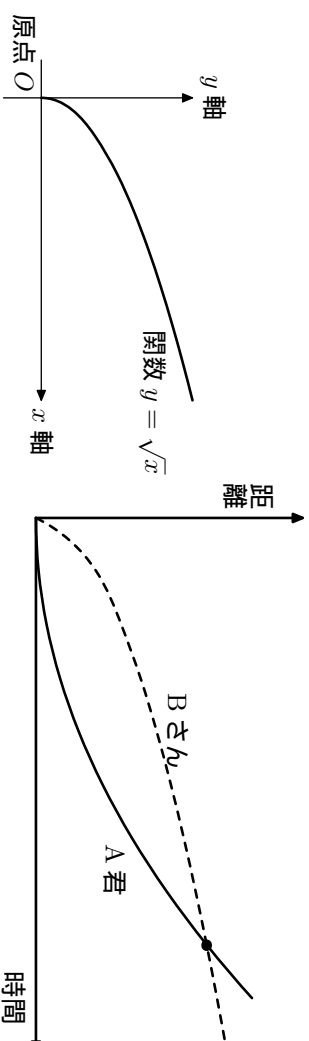


で、のようになる。usexobjでは同じ図を何度も取り込まなくて済むが多少扱いが難しく、使う人はあまりいないと思われるので、これではある。

2 日本語 METAPOST 対応

dvipdfm に日本語 METAPOST 対応を加えてみた。日本語 METAPOST に関する情報は <http://www.sat.t.u-tokyo.ac.jp/%7Ehideyuki/metapost/> を参照して頂きたい。対応といっても些細な変更で済んだ。

日本語 METAPOST では、図のようにレベルに日本語を使うことができる。



図一 日本語 METAPOST の出力例。縦組みなどの図は回転して表示されている。

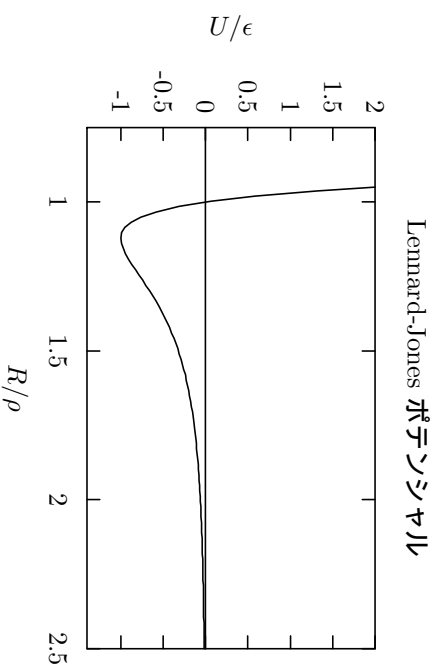
METAPOST の出力は通常の PostScript ファイルとは異なり dvipdfm が直接 PDF に変換する。METAPOST が生成する PostScript は構造が比較的単純なため、dvipdfm が直接 PDF に変換して取り込むことができる。図を見て頂けると分かると思うが、フォントは TeX で使われているフォントと同じように処理される。これは METAPOST を使う利点の一つである。METAPOST 出力をサポートしているアプリケーションは少ないが、TeX の連携を考へるのならば METAPOST が理想的な選択肢。prologues:=1 にすると dvipdfm で処理できないが、これはしょうがない。



最近の Gnuplot と METAPOST 出力をサポートしている。試すよ

```
gnuplot> set term mp "rml" 10
```

とじつ METAPOST で出力してみた。



ちなみに **META**Graf でも日本語が使えた。

3 おわりに

縦組みが必要なのは文系の人や国語教師ということになるだろう。残念ながらこれらの人達にとってはあまり使いものにならないのかもしれない。基本的にJS 第一・第二水準の漢字しかサポートしていないからだ。しかしながら、私は dvipdfm の日本語化はこれで基礎的な部分は完了したと考えている。理工系のレポートや論文などを作成したり、ちょっとしたマニュアルを PDF で配布するといった用途では特に困る事はないだろう。中国語や韓国語などを含め、より多くの文字を CID フォントとして使えるようにできればよいと思うが、これを行なうにはもっと積極的に dvipdfm に手を加えていく必要がある。和文フォントの埋め込みなど、より完全な日本語サポートを目指した取り組みは既に行なわれているが、「より完全なものにしたい」ということ以外に動機がないため開発のペースは非常に遅い。

いろいろとテストしてみたいという方は <http://imt.chem.kit.ac.jp/fujita/fujitas/RD/textlatex.html> にある DVI ファイルを変換してみるとよいかもしれない。化学構造式や漢文の訓点文などの例があってなかなか面白い。