

Package `bounddvi` v7.1

Koichi Inoue & Hironobu Yamashita

2016/10/25

DVI ファイルは用紙サイズ情報を含みませんが、PostScript や PDF といった用紙サイズを規定するフォーマットへ変換するプログラム (`dvips` や `dvipdfmx`) は、`\special` 命令によって DVI ファイルに埋め込まれた用紙サイズ情報を解釈します。

`bounddvi` パッケージは、クラスファイルなどが想定している用紙サイズ情報 (`\paperwidth`, `\paperheight`) を DVI ファイルに `\special` 命令として埋め込むことで、`dvips` や `dvipdfmx` の出力用紙サイズを指定する機能を提供します。

```
\documentclass[a4j]{jarticle}
\usepackage{bounddvi}
\begin{document}
…本文…
\end{document}
```

のように使います。pL^AT_EX における横組と縦組の両方で使え、また版面拡大 (`\mag`) が使われた場合にも対応しています (詳細は付録を参照してください)。また、v7.0 以降、pL^AT_EX カーネルのトンボオプションを使用した場合にも対応しており、pL^AT_EX が仮定している「トンボのノビ (2in) を含めたサイズ」に設定します。

本パッケージは v7.0 以降、`platex-tools` バンドルの一部として配布されています：

<https://github.com/aminophen/platex-tools>

他のパッケージとの互換性

用紙サイズ設定機能だけを提供する `bounddvi` パッケージのほかにも、パッケージの機能の一部として用紙サイズ設定を行うものがあります。

- 日本語環境でよく用いられる、pL^AT_EX/upL^AT_EX 用のクラスファイル `jsclasses` (奥村さん作) の `papersize` オプションとの併用は問題ありません (ただし、このオプションを使用すれば用紙サイズ設定には十分ですから、本 `bounddvi` パッケージを併用することにも特にメリットはありません)。
- 2016 年以降の `graphics/graphicx` パッケージや `color` パッケージには用紙サイズ設定機能 (`setpagesize` オプション：デフォルトで有効) がありますが、併用しても問題は起きないでしょう。
- `hyperref` パッケージも上と同様で、併用しても問題は起きないでしょう。
- `geometry` パッケージと併用する場合は、本 `bounddvi` パッケージを後から読み込むほうが安全でしょう。特に、`dvipdfmx` を使用して変換する場合にトンボを付けたいなら、`bounddvi` を後から読み込むこ

とどうまくいく可能性があります (付録を参照)。

付録：dvips と dvipdfmx の違い

一つの DVI ファイルに複数の用紙サイズ指定 `\special` 命令が埋め込まれる場合があります。このうち、`dvips` では DVI ファイルの最初に指定されたものが有効になります。逆に `dvipdfmx` では、DVI ファイルの最後に指定されたものが有効になります。このため、たとえば `bounddvi` と `geometry` を併用したソース

```
\documentclass{...}
\usepackage{bounddvi}
\usepackage[dvipdfm]{geometry}
```

を `dvipdfmx` で PDF に変換すると、用紙サイズの設定が正しくなされない可能性があります。

付録：`\mag` の扱いについて

本 `bounddvi` パッケージは、日本で広く使われている奥村さんの `jsclasses` クラス流の `\mag` の用法をサポートしています。しかし、`\mag` の使い方は CTAN にある多様なパッケージの中で統一されておらず、大きく分けて二通りの使い方があるようです。

たとえば、物理的な A4 サイズ (210 mm × 297 mm) の出力を得たい場合

```
\mag=2000
\paperwidth=210mm (= 420truemm)
\paperheight=297mm (= 594truemm)
```

と設定するものと

```
\mag=2000
\paperwidth=105mm (= 210truemm)
\paperheight=148.5mm (= 297truemm)
```

と設定するものが存在します。前者は `geometry` パッケージなどの方式で、おそらく `dvips` の `papersize special` の挙動 (`truedimen` を正しく解釈せず、`true` の付かない単位だけを受け付けてあたかも `true` 付であるかのように評価) に則ったものでしょう。後者は `jsclasses` クラスの方式で、これは `dvipdfm(x)` の `pdf:pagesize special` と同じです。この方式のほうが、`truedimen` で設定された \LaTeX のほかのレイアウト変数 (`\textwidth` など) に対して一貫性があると思われます。本 `bounddvi` パッケージがサポートしているのは後者の方式です。

参考

- `dvips` や `dvipdfm` などでの用紙サイズを出力
<https://www.ma.ns.tcu.ac.jp/Pages/TeX/bounddvi.sty.html>