

dvipng

A DVI-to-PNG Translator
Version 1.5

by Jan-Åke Larsson.

Copyright © 2002-2005 Jan-Åke Larsson

Permission is granted to make and distribute verbatim copies of this manual provided the copyright notice and this permission notice are preserved on all copies.

Permission is granted to copy and distribute modified versions of this manual under the conditions for verbatim copying, provided also that the section entitled “Copying” is included exactly as in the original, and provided that the entire resulting derived work is distributed under the terms of a permission notice identical to this one.

Permission is granted to copy and distribute translations of this manual into another language, under the above conditions for modified versions, except that this permission notice may be stated in a translation approved by the Free Software Foundation.

Table of Contents

dvipng	1
1 Introduction	2
2 Installation	3
2.1 Prerequisites	3
2.2 Configure	3
2.3 Build/install	4
2.4 Installation outside the texmf tree	4
2.5 Installation for non-privileged users	4
3 Basic usage of dvipng	6
4 Command-line options	7
4.1 Option summary	7
4.2 Option details	8
5 Diagnosing problems	13
5.1 Debug options	13
5.2 Included graphics problems	14
6 Color	15
6.1 Color specifications	15
6.2 Color specials	15
7 Copying	16
Index	17

dvipng

1 Introduction

This program makes PNG and/or GIF graphics from DVI files as obtained from \TeX and its relatives.

If GIF support is enabled, GIF output is chosen by using the ‘`dvigif`’ binary or with the ‘`--gif`’ option.

It is intended to produce anti-aliased screen-resolution images as fast as is possible. The target audience is people who need to generate and regenerate many images again and again. The primary target is the Preview- \LaTeX (X)Emacs package, a package to preview formulas from within (X)Emacs. Yes, you get to see your formulas in the (X)Emacs buffer, see <http://preview-latex.sf.net>.

Another example is WeBWorK, an internet-based method for delivering homework problems to students over the internet, giving students instant feedback as to whether or not their answers are correct, see <http://webwork.math.rochester.edu>.

Other applications may also benefit, like web applications as latex2html and WYSIWYG editors like LyX.

The benefits of ‘`dvipng`’/‘`dvigif`’ include

- Speed. It is a very fast bitmap-rendering code for DVI files. On a fairly low-end laptop, it takes less than a second to generate 150 one-formula images. This makes dvipng suitable for generating large amounts of images on-the-fly, as needed in Preview- \LaTeX , WeBWorK and others.
- It does not read the postamble, so it can be started before \TeX finishes. There is a `--follow` switch that makes dvipng wait at EOF for further output, unless it finds the POST marker that indicates the end of the DVI.
- Interactive query of options. dvipng can read options interactively through stdin, and all options are usable. It is even possible to change the input file through this interface.
- Supports PK, VF, PostScript Type1, and TrueType fonts, color specials and simple PostScript inclusion specials.
- and more...

2 Installation

Installing dvipng should be simple: merely `./configure`, `make`, and `make install`.

2.1 Prerequisites

- The GD Graphics Draw library, `libgd`

The drawing library ‘`libgd`’ is necessary, and is downloadable at <http://www.boutell.com/gd>, and there are binary packages for most operating systems from their respective distributors. In any case, the latest version of the library installs using ‘`autoconf`’ so it should not be difficult for you to install it from source, and then proceed with installing dvipng.

- The path-searching library `kpathsea`

Kpathsea is most likely included in your L^AT_EX installation, but it may happen that `./configure` does not find it; see below. If you do not have it, download it from <http://www.ctan.org> and compile it. I have no experience with this, so I cannot help much here.

- FreeType 2 and/or T1lib

While not strictly necessary, you may want to have FreeType 2 or T1lib available. Their presence will enable direct support for PostScript fonts in dvipng, rather than using ‘`gsftopk`’ for rendering the fonts. If you have PostScript versions of Computer Modern installed, there will be no need to generate bitmapped variants on disk of these. Then, you can render images at different (and unusual) resolutions without cluttering the disk with lots of bitmapped fonts. If both FreeType 2 and T1lib are present, FreeType will be internally preferred by dvipng, but T1lib can be chosen at runtime.

- `libpng` and `libz`

To be able to compress and write PNG files to disk, dvipng (or really `libgd`) uses `libpng` which in turn uses `libz`. These should be available on any modern system, if not, download them and install them.

- The `texinfo` package

This is needed for building the documentation.

2.2 Configure

The first step is to configure the source code, telling it where various files will be. To do so, run

```
./configure options
```

(Note: if you have fetched dvipng from CVS rather than a regular release, you will have to first generate ‘`./configure`’ by running `autoconf 2.53` or later in the ‘`preview`’ directory.)

On many machines, you will not need to specify any options, but if `configure` cannot determine something on its own, you’ll need to help it out. For a list of the options type

```
./configure --help
```

On some machines, the libraries will be installed in directories that are not in the linker’s search path. This will generate an error when running ‘`./configure`’, indicating that it

cannot find `libgd` or `libkpathsea` (most likely). You then need to specify the path to the respective library's object files. They are typically called e.g., `'libgd.a'` or `'libgd.so'`. If they are located in e.g., `'/sw/local/lib'`, do

```
./configure LDFLAGS=-L/sw/local/lib
```

If the library is available as a shared object file (`'so'`), the runtime linker may also need to be told where to find the library, then use

```
./configure LDFLAGS='-L/sw/local/lib -R/sw/local/lib'
```

When either of these is necessary, it is likely that the C header files are also installed in directories that are not in the C preprocessor's search path. This will also generate an error when running `'./configure'`, indicating that it cannot find e.g., `'gd.h'` or `'kpathsea.h'` (most likely). You then need to specify the path to the respective library's C header files. If they are located in e.g., `'/sw/local/include'`, do

```
./configure CPPFLAGS=-I/sw/local/include
```

On my SUN Solaris workstation, I had to combine this into

```
./configure CPPFLAGS='-I/sw/local/include -I/sw/tex/teTeX/1.0/include'\
LDFLAGS='-L/sw/local/lib -R/sw/local/lib -L/sw/tex/teTeX/1.0/lib/'
```

where the backslash denotes a continuation of the line.

2.3 Build/install

Once `'configure'` has been run, simply enter

```
make
```

at the prompt to compile the C code, and build the documentation files. To install the files into the locations chosen earlier, type

```
make install
```

You may need special privileges to install, e.g., if you are installing into system directories.

2.4 Installation outside the texmf tree

In some cases, a `dvipng` binary installed outside the `texmf` tree will not be able to find virtual fonts, or the PostScript font maps (normally used by `dvips`). This may be because *only* `$SELFAUTOLOC`, `$SELFAUTODIR`, and `$SELFAUTOPARENT` are used in the `texmf` tree configuration file `'texmf.cnf'`. If so, give the switch `'--enable-selfauto-set'` to `'./configure'`. This will make `dvipng` adjust these three internally so that `kpathsea` thinks that `dvipng` *is* installed in the `texmf` tree.

2.5 Installation for non-privileged users

Often people without system administration privileges want to install software for their private use. In that case you need to specify more options to the `'configure'` script, usually this is done by using the `'--prefix'` option to the `'configure'` script, and let it point to the personal home directory. In that way, resulting binaries will be installed under the `'bin'` subdirectory of your home directory, manual pages under `'man'` and so on. That way, it is reasonably easy to maintain a bunch of additional packages, since the prefix argument is supported by most `'configure'` scripts.

You'll have to add something like `‘/home/myself/bin’` to your `PATH` shell variable, if it isn't there already, and similarly set the `INFOPATH` variable to be able to access the documentation.

3 Basic usage of dvi2png

To use dvi2png at its simplest, simply type

```
dvi2png foo
```

where ‘foo.dvi’ is the output of \TeX that you want to convert to PNG format. If there are four pages in ‘foo.dvi’, those pages will be output as ‘foo1.png’, ‘foo2.png’, ‘foo3.png’, and ‘foo4.png’, respectively.

If you have enabled the PostScript font support (via FreeType or T1lib), fonts will be rendered as they are needed. Otherwise, if you use fonts that have not been used on your system before, they may be automatically generated; this process can take a few minutes, so progress reports appear by default. The next time the same font is used, it will have been saved on disk, so rendering will go much faster. (If dvi2png tries to endlessly generate the same fonts over and over again, something is wrong. See [section “Unable to generate fonts” in *Kpathsea*](#).)

Many options are available (see the next section). For a brief summary of available options, just type

```
dvi2png --help
```

4 Command-line options

dvipng has a plethora of command line options. Reading through this section will give a good idea of the capabilities of the driver.

4.1 Option summary

Here is a handy summary of the options; it is printed out when you run dvipng with no arguments or with the standard ‘--help’ option.

This is ./dvipng 1.2 Copyright 2002-2004 Jan-ke Larsson

Usage: ./dvipng [OPTION]... FILENAME[.dvi]

Options are chosen to be similar to dvips’ options where possible:

-d #	Debug (# is the debug bitmap, 1 if not given)
-D #	Output resolution
-l #	Last page to be output
--mode s	MetaFont mode (default ‘cx’)
-M*	Don’t make PK fonts
-o f	Output file, ‘%d’ is pagenummer
-O c	Image offset
-p #	First page to be output
-pp #,#..	Page list to be output
-q*	Quiet operation
-r*	Reverse order of pages
-T c	Image size (also accepts ‘-T bbox’ and ‘-T tight’)
-v*	Verbose operation
-x #	Override dvi magnification
-	Interactive query of options

These do not correspond to dvips options:

-bd #	Transparent border width in dots
-bdpi #	Set the base (Metafont) resolution
-bg s	Background color (TeX-style color)
--depth*	Output the image depth on stdout
--dvinum*	Use TeX page numbers in output filenames
-fg s	Foreground color (TeX-style color)
--follow*	Follow mode
--gif	Output GIF images (dvigif default)
--height*	Output the image height on stdout
--png	Output PNG images (dvipng default)
--tlib*	Ttlib font rendering (default on)
--truecolor*	Truecolor output
-Q #	Quality (Ttlib and PK subsampling)
-z #	PNG compression level

= number f = file s = string * = suffix, ‘0’ to turn off
c = comma-separated dimension pair (e.g., 3.2in,-32.1cm)

4.2 Option details

Many of the parameterless options listed here can be turned off by suffixing the option with a zero ('0'); for instance, to turn off page reversal, use '-r0'. Such options are marked with a trailing '*'.

'-' Read additional options from standard input after processing the command line.

'--help' Print a usage message and exit.

'--version' Print the version number and exit.

'-bd *num*'

'-bd *color_spec*'

'-bd '*num color_spec*'

Set the pixel width of the transparent border (default 0). Using this option will make the image edges transparent, but it only affects pixels with the background color. Giving a *color_spec* will set the fallback color, to be used in viewers that cannot handle transparency (the default is the background color). The color spec should be in T_EX color \special syntax, e.g., 'rgb 1.0 0.0 0.0'. Setting the fallback color makes the default border width 1 px. See [Chapter 6 \[Color\]](#), [page 15](#).

'--bdpi *num*'

Set the base (Metafont) resolution, both horizontal and vertical, to *num* dpi (dots per inch). This option is necessary when manually selecting Metafont mode with the -mode option (see below).

'-bg *color_spec*'

Choose background color for the images. This option will be ignored if there is a background color \special in the DVI. The color spec should be in T_EX color \special syntax, e.g., 'rgb 1.0 0.0 0.0'. You can also specify 'Transparent' which will give you a transparent background with the normal background as a fallback color. See [Chapter 6 \[Color\]](#), [page 15](#).

'-d *num*' Set the debug flags, showing what dvi_{png} (thinks it) is doing. This will work unless dvi_{png} has been compiled without the 'DEBUG' option (not recommended). Set the flags as you need them, use '-d -1' as the first option for maximum output. See [Section 5.1 \[Debug options\]](#), [page 13](#).

'-D *num*' Set the output resolution, both horizontal and vertical, to *num* dpi (dots per inch).

One may want to adjust this to fit a certain text font size (e.g., on a web page), and for a text font height of *font_px* pixels (in Mozilla) the correct formula is

$$dpi = font_px * 72.27 / 10 \text{ [px * T_EXpt/in / T_EXpt]}$$

The last division by ten is due to the standard font height 10pt in your document, if you use 12pt, divide by 12. Unfortunately, some proprietary browsers

have font height in pt (points), not pixels. You have to rescale that to pixels, using the screen resolution (default is usually 96 dpi) which means the formula is

$$\text{font_px} = \text{font_pt} * 96 / 72 \text{ [pt} * \text{px/in} / \text{(pt/in)]}$$

On some high-res screens, the value is instead 120 dpi. Good luck!

`--depth*`

Report the depth of the image. This only works reliably when the L^AT_EX style ‘`preview.sty`’ from Preview-L^AT_EX is used. It reports the number of pixels from the bottom of the image to the baseline of the image. This can be used for vertical positioning of the image in, e.g., web documents, where one would use (Cascading StyleSheets 1)

```
<IMG SRC="filename.png" STYLE="vertical-align: -depthpx">
```

The depth is a negative offset in this case, so the minus sign is necessary, and the unit is pixels (px).

`--dvinum*`

Set this option to make the output page number be the T_EX page numbers rather than the physical page number. See the ‘`-o`’ switch.

`-fg color_spec`

Choose foreground color for the images. This option will be ignored if there is a foreground color `\special` in the DVI. The color spec should be in T_EX color `\special` syntax, e.g., ‘`rgb 1.0 0.0 0.0`’. See [Chapter 6 \[Color\]](#), [page 15](#).

`--follow*`

Enable follow mode. One of the benefits of `dvipng` is that it does not read the postamble, so it can be started before T_EX finishes. This switch makes `dvipng` wait at EOF for further output, unless it finds the POST marker that indicates the end of the DVI. This is similar to ‘`tail -f`’ but for DVI-to-PNG conversion.

`--freetype*`

Enable/disable FreeType font rendering (default on). This option is available if the FreeType2 font library was present at compilation time. If this is the case, `dvipng` will have direct support for PostScript Type1 and TrueType fonts internally, rather than using ‘`gsftopk`’ for rendering the fonts. If you have PostScript versions of Computer Modern installed, there will be no need to generate bitmapped variants on disk of these. Then, you can render images at different (and unusual) resolutions without cluttering the disk with lots of bitmapped fonts. Note that if you have both FreeType and T1lib on your system, FreeType will be preferred by `dvipng`. If you for some reason would want to use T1lib rendering, use this option.

`--gamma num`

Control the interpolation of colors in the greyscale anti-aliasing color palette. Default value is 1.0. For $0 < \text{num} < 1$, the fonts will be lighter (more like the background), and for $\text{num} > 1$, the fonts will be darker (more like the foreground).

- `--gif*` The images are output in the GIF format, if GIF support is enabled. This is the default for the `dvigif` binary, which only will be available when GIF support is enabled. See also the `--png` option.
- `--height*` Report the height of the image. This only works reliably when the L^AT_EX style `preview.sty` from Preview-L^AT_EX is used. It reports the number of pixels from the top of the image to the baseline of the image. The total height of the image is obtained as the sum of the values reported from `-height` and the `-depth`.
- `-l [=]num` The last page printed will be the first one numbered *num*. Default is the last page in the document. If *num* is prefixed by an equals sign, then it (and the argument to the `-p` option, if specified) is treated as a physical (absolute) page number, rather than a value to compare with the T_EX `\count0` values stored in the DVI file. Thus, using `-l =9` will end with the ninth page of the document, no matter what the pages are actually numbered.
- `--mode mode` Use *mode* as the Metafont device name for the PK fonts (both for path searching and font generation). This needs to be augmented with the base device resolution, given with the `--bdpi` option. See the file <ftp://ftp.tug.org/tex/modes.mf> for a list of resolutions and mode names for most devices. See [section “Unable to generate fonts” in Kpathsea](#).
- `-M*` Turns off automatic PK font generation (`mktexpk`). This will have no effect when using PostScript fonts, since no PK font generation will be done anyway.
- `--noghostscript*` This switch prohibits the internal call to GhostScript for displaying PostScript specials. `--noghostscript0` turns the call back on.
- `--nogssafer*` Normally, if GhostScript is used to render PostScript specials, the GhostScript interpreter is run with the option `-dSAFER`. The `--nogssafer` option runs GhostScript without `-dSAFER`. The `-dSAFER` option in Ghostscript disables PostScript operators such as `deletfile`, to prevent possibly malicious PostScript programs from having any effect.
- `-o name` Send output to the file *name*. A single occurrence of `%d` or `%01d`, . . . , `%09d` will be exchanged for the physical page number (this can be changed, see the `--dvinum` switch). The default output filename is `file%d.png` where the input DVI file was `file.dvi`.
- `-O x-offset,y-offset` Move the origin by *x-offset,y-offset*, a comma-separated pair of dimensions such as `.1in,-.3cm`. The origin of the page is shifted from the default position (of one inch down, one inch to the right from the upper left corner of the paper) by this amount.

`'-p [=]num'`

The first page printed will be the first one numbered *num*. Default is the first page in the document. If *num* is prefixed by an equals sign, then it (and the argument to the `'-l'` option, if specified) is treated as a physical (absolute) page number, rather than a value to compare with the T_EX `'\count0'` values stored in the DVI file. Thus, using `'-p =3'` will start with the third page of the document, no matter what the pages are actually numbered.

`'--picky'`

No images are output when a warning occurs. Normally, dvipng will output an image in spite of a warning, but there may be something missing in this image. One reason to use this option would be if you have a more complete but slower fallback converter. Mainly, this is useful for failed figure inclusion and unknown `\special` occurrences, but warnings will also occur for missing or unknown color specs and missing PK fonts.

`'--png'` The images are output in the PNG format. This is the default for the `'dvipng'` binary. See also the `'--gif'` option.

`'-pp firstpage-lastpage'`

Print pages *firstpage* through *lastpage*; but not quite equivalent to `'-p firstpage -l lastpage'`. For example, when rendering a book, there may be several instances of a page in the DVI file (one in `\frontmatter`, one in `\mainmatter`, and one in `\backmatter`). In case of several pages matching, `'-pp firstpage-lastpage'` will render *all* pages that matches the specified range, while `'-p firstpage -l lastpage'` will render the pages from the *first* occurrence of *firstpage* to the *first* occurrence of *lastpage*. This is the (undocumented) behaviour of dvips. In dvipng you can give both kinds of options, in which case you get all pages that matches the range in `'-pp'` between the pages from `'-p'` to `'-l'`. Also multiple `'-pp'` options accumulate, unlike `'-p'` and `'-l'`. The `'-'` separator can also be `':'`. Note that `'-pp -1'` will be interpreted as "all pages up to and including 1", if you want a page numbered -1 (only the table of contents, say) put `'-pp -1--1'`, or more readable, `'-pp -1:-1'`.

`'-q'` Run quietly. Don't chatter about pages converted, etc. to standard output; report no warnings (only errors) to standard error.

`'-Q num'` Set the quality to *num*. That is, choose the number of antialiasing levels for PK and T1lib rendering to be *num***num*. The default value is 4 which gives 16 levels of antialiasing for antialiased fonts from these two. If FreeType is available, its rendering is unaffected by this option.

`'-r'` Toggle output of pages in reverse/forward order. By default, the first page in the DVI is output first.

`'-T image_size'`

Set the image size to *image_size* which can be either of `'bbox'`, `'tight'`, or a comma-separated pair of dimensions *hsize*,*vsize* such as `'.1in,.3cm'`. The default is `'bbox'` which produces a PNG that includes all ink put on the page and in addition the DVI origin, located 1in from the top and 1in from the left edge of the paper. This usually gives whitespace above and to the left in the

produced image. The value `'tight'` will make `dvipng` only include all ink put on the page, producing neat images. This option overrides any `papersize` special in the DVI file.

`'--t1lib*`

Enable/disable T1lib font rendering (default on). This option is available if the T1lib font library was present at compilation time. If this is the case, `dvipng` will have direct support for PostScript Type1 fonts internally, rather than using `'gsftopk'` for rendering the fonts. If you have PostScript versions of Computer Modern installed, there will be no need to generate bitmapped variants on disk of these. Then, you can render images at different (and unusual) resolutions without cluttering the disk with lots of bitmapped fonts. Note that if you have both FreeType and T1lib on your system FreeType will be preferred by `dvipng`, and if you for some reason rather want to use T1lib, give the option `'--freetype0'` (see above).

`'--truecolor*`

On systems with a fairly new `'libgd'`, one can choose to generate truecolor output. This will not be necessary in general unless you include truecolor images via a PostScript special (i.e., the `'graphics'` or `'graphicx'` package). It will lead to longer execution time, mostly because more data is written to disk.

`'-v*`

Enable verbose operation. This will currently indicate what fonts is used, in addition to the usual output.

`'-x num'`

Set the x magnification ratio to $num/1000$. Overrides the magnification specified in the DVI file. Must be between 10 and 100000. It is recommended that you use standard magstep values (1095, 1200, 1440, 1728, 2074, 2488, 2986, and so on) to help reduce the total number of PK files generated. num may be a real number, not an integer, for increased precision.

`'-z num'`

Set the PNG compression level to num . This option is enabled if your `'libgd'` is new enough. The default compression level is 1, which selects maximum speed at the price of slightly larger PNGs. For an older `'libgd'`, the hard-soldered value 5 is used. The include file `'png.h'` says

Currently, valid values range from 0 - 9, corresponding directly to the zlib compression levels 0 - 9 (0 - no compression, 9 - "maximal" compression). Note that tests have shown that zlib compression levels 3-6 usually perform as well as level 9 for PNG images, and do considerably fewer calculations. In the future, these values may not correspond directly to the zlib compression levels.

5 Diagnosing problems

You’ve gone through all the trouble of installing dvipng, carefully read all the instructions in this manual, and still can’t get something to work. The following sections provide some helpful hints if you find yourself in such a situation.

Bug reports should be sent to dvipng-devel@lists.sourceforge.net.

Questions, suggestions for new features, pleas for help, and/or praise should go to dvipng-devel@lists.sourceforge.net (the dvipng devel team). For more information on this mailing list, send a message with just the word ‘help’ as subject or body to dvipng-devel-request@lists.sourceforge.net or look at <http://lists.sourceforge.net/lists/listinfo/dvipng-devel>.

Offers to support further development will be appreciated. For developer access, ask on dvipng-devel@lists.sourceforge.net.

For details on the T_EX path-searching library, and `mktexpk` problems, see [section “Common problems” in *Kpathsea*](#).

5.1 Debug options

The ‘-d’ flag to dvipng helps in tracking down certain errors. The parameter to this flag is an integer that tells what errors are currently being tracked. To track a certain class of debug messages, simply provide the appropriate number given below; if you wish to track multiple classes, sum the numbers of the classes you wish to track. To track all classes, you can use -1.

Some of these debugging options are actually provided by Kpathsea (see [section “Debugging” in *Kpathsea*](#)).

The classes are:

1	Normal dvi op-codes
2	Virtual fonts
4	PK fonts
8	TFM files
16	Glyph rendering
32	FreeType calls
64	Encoding loads
128	Color specials
256	GhostScript specials
512	T1lib calls
1024	Kpathsea <code>stat</code> calls
2048	Kpathsea hash table lookups
4096	Kpathsea path element expansion
8192	Kpathsea path searches

5.2 Included graphics problems

The most common problem with including graphics is an incorrect bounding box. Complain to whoever wrote the software that generated the file if the bounding box is indeed incorrect.

An additional problem is that the image in dvipng is always clipped to the bounding box. This is because the image is rendered by GhostScript by an internal call and then copied to the dvipng image. At present no workaround is known, but there may be one for a future GhostScript. We'll see.

6 Color

To support color, dvipng recognizes a certain set of specials. These specials start with the keyword `'color'` or the keyword `'background'`, followed by a color specification.

6.1 Color specifications

The color specification supported by dvipng is the name of a color model (initially, one of `'rgb'`, `'hsv'`, `'cmyk'`, or `'gray'`) followed by the appropriate number of parameters. Thus, the color specification `'rgb 0.3 0.4 0.5'` would correspond to the color that is `'0.3 0.4 0.5'` in its red, blue and green values.

6.2 Color specials

We will describe `'background'` first, since it is the simplest. The `'background'` keyword must be followed by a color specification. That color specification is used as a fill color for the background. The last `'background'` special on a page is the one that gets used, and is used for the whole of the page image. (This is possible because the prescan phase of dvipng notices all of the color specials so that the appropriate information can be written out during the second phase.)

The `'color'` special itself has three forms. The first is just `'color'` followed by a color specification. In this case, the current global color is set to that color; the color stack must be empty when such a command is executed.

The second form is `'color push'` followed by a color specification. This saves the current color on the color stack and sets the color to be that given by the color specification. This is the most common way to set a color.

The final version of the `'color'` special is just `'color pop'`, with no color specification; this says to pop the color last pushed on the color stack from the color stack and set the current color to be that color.

dvipng correctly handles these color specials across pages, even when the pages are rendered repeatedly or in reverse order.

7 Copying

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111-1307, USA.

Copyright © 2002-2005 Jan-Åke Larsson

Index

-

-dSAFER 10

A

absolute page number, and ‘-l’ 10

absolute page number, and ‘-p’ 11

antialiasing levels, number of 11

B

background color (option) 8

base resolution, setting 8

baseline reporting 9, 10

C

color specifications 15

command-line options 7

compilation 3

compression 12

configuration, of dvipng 3

D

dark fonts 9

debugging 8, 13

debugging options 13

depth reporting 9

F

first page printed 11

follow mode 9

font generation, avoiding 10

foreground color (option) 9

FreeType font rendering 9

fuzzy images 9

G

gamma 9

GhostScript and -dSAFER 10

GhostScript, turning off 10

GIF image format 10

graphics inclusion problems 14

H

height reporting 10

I

installation, of dvipng 3

invoking dvipng 6

W

L

last page printed 10

light fonts 9

M

magnification, overriding DVI 12

Metafont mode, specifying 10

mk texpk , avoiding 10

mode name, specifying 10

N

no erroneous images 11

O

offset pages 10

option, details of 8

options, debugging 13

options, dvipng 7

options, reading from standard input 8

options, summary 7

output resolution, setting 8

output, redirecting 10

P

page range 11

page, first printed 11

page, last printed 10

physical page number, and ‘-l’ 10

physical page number, and ‘-p’ 11

PNG image format 11

problems 13

Q

quality 11

quiet operation 11

R

reverse pagination 11

S

silent operation 11

standard input, reading options from 8

standard output, output to 10

T

T tlib font rendering 12

transparent border fallback color 8

transparent border width 8

trouble 13

truecolor output 12