

The KDVI Handbook

Stefan Kebekus



The KDVI Handbook

Contents

1	Introduction	1
2	Starting KDVI	2
3	Printing DVI Files	3
4	Exporting the DVI file to other formats	4
4.1	Exporting to PostScript®	4
4.2	Exporting to PDF	4
4.3	Exporting to text files	5
5	Embedding PostScript files into the DVI	6
6	Using inverse search	7
6.1	Producing TeX files for inverse search	7
6.2	Setting up your editor for inverse search	8
6.2.1	Emacs	8
6.2.2	Kate	11
6.2.3	Kile	11
6.2.4	NEdit	11
6.2.5	XEmacs	11
6.2.6	VI iMproved / GUI	12
7	Forward search	13
7.1	Setting up your editor for forward search	13
7.1.1	Emacs	13
7.1.2	Kile	14
7.1.3	XEmacs	14

The KDVI Handbook

8	The Preferences dialog	15
8.1	Fonts Options	15
8.2	DVI specials Options	16
9	Frequently asked questions	18
10	Credits and Licenses	19

Abstract

This document describes KDVI version 1.1

Chapter 1

Introduction

KDVI is a plugin for the KViewshell program which allows KViewshell to display DVI-files (`.dvi`) which are produced by the TeX typesetting system. KDVI supports many extensions of the DVI standard, for instance the inclusion of PostScript® graphics or hyperlinks. More details, examples and all the technical specifications can be found in the file `KDVI-features.dvi` (or see `KDVI-features.tex` for the TeX source of that file).

For up-to-date information, consult [KDVI's home page](#).

TeX is a high-end typesetting system geared towards scientific, and in particular mathematical typesetting. More information about TeX and DVI can be found on the [homepage of the TeX user group](#) or the German [German DANTE e.V.](#)

Chapter 2

Starting KDVI

Most of the time, KDVI will be started by just clicking onto a `.dvi` file in the file manager. For convenience there exists a command `kdvi` which calls KViewshell with the KDVI plugin preloaded. The viewer may thus be started using the command `kdvi somepath/paper.dvi`. The command lines `kdvi somepath/paper` or `kdvi somepath/paper.` will also work. If you are connected to the internet, you can access files which reside on other computers by giving a URL as a parameter, like this: `kdvi http://somepath/paper.dvi`

If you give a URL as a parameter, you can tell KDVI to jump directly to certain place of the DVI file. For example, `kdvi file:paper.dvi#43` will make KDVI to open page 43. If you have included source file information, a command like `kdvi file:paper.dvi#src:43paper.tex` will make KDVI search for the place in the DVI file which corresponds to line 43 in the TeX file `paper.tex`. You will hardly use this option yourself — read the section on [forward search](#) to learn how to set up your editor to start KDVI automatically.

WARNING

Don't forget the `file:` prefix or it will give unexpected results. For example, the command `kdvi file:paper.dvi#43` will open page 43 of the file `paper.dvi`. The command `kdvi paper.dvi#43` will try to open the file `paper.dvi#43`.

There is another option which you will most likely not need to specify yourself. If you type `kdvi --unique somepath/paper.dvi`, KDVI will load the file if there is no other instance running which has the file already loaded. If there is, this instance of KDVI will pop to the front. A command like `kdvi --unique file:paper.dvi#43` can be used in shell scripts to make a running instance of KDVI to jump to page 43.

The usual parameters handled by QtTM and KDE applications also work: `kdvi -style windows -display :0 -geometry 400x400+0+0 -caption "DVI"`

Chapter 3

Printing DVI Files

KDVI can print your DVI files using the standard KDE printing interface. Internally, KDVI uses the program **dvips** to generate PostScript®, which is then passed on to the printer. In particular, **dvips** must be installed if you want to print with KDVI. The program **dvips** uses its own configuration files and its own settings, which are fine for most purposes. However, if you care for optimal printing results, you should configure **dvips** manually and make sure to set a default MetaFont mode which fits your printer best — on many systems you'll find a [GNU-texinfo documentation of dvips](#), and you might also want to look for a file called `dvips.dvi` or similar.

Chapter 4

Exporting the DVI file to other formats

If you want to save your file as in PostScript® or PDF format, it is not recommended that you use the printing function and redirect the printer output to a file. Instead, you can use the export functions which produce better-quality output that retains many of the special features of the dvi format and looks better in many of the viewing applications, such as Adobe's Acrobat Reader. You will find the export functions in the File menu.

4.1 Exporting to PostScript®

As in printing, the external program **dvips** is used to generate the PostScript® file. If the DVI file contains hyperlinks, these will also be included in the PostScript® file. If you are an expert, and if you would like to generate output which is optimized for a specific printer, you should probably start **dvips** manually and choose the proper MetaFont mode yourself.

4.2 Exporting to PDF

In order to produce PDF files of high quality, KDVI converts DVI to PDF using the external program **dvipdfm**. If you are working on a machine where an older distribution of the TeX typesetting system is installed, it may be that the program **dvipdfm** is not installed. In that case, you need to use the printing function to generate PDF output.

WARNING

- If you use an older TeX installation, and if are viewing the generated file in Adobe's Acrobat reader, you may well find that some of the fonts look extremely poor although a printout is fine, and although the document looks ok in **kghostview**. This is a known issue with the Acrobat Reader and bitmap fonts. At the time of writing, the only practicable workaround seems to be to avoid bitmap fonts, or to upgrade to a more recent TeX installation.
- While **dvipdfm** produces high-quality PDF files, **dvipdfm** currently currently ignores the PostScript® that is embedded into the DVI file. Embedded PostScript is generated e.g. by the xy macro package, or by the "Embed PostScript files" function KDVI described [below](#).
If you find that the generated PDF file misses graphical data, use the print function of KDVI instead.

4.3 Exporting to text files

KDVI can also save your DVI files in text format.

WARNING

The DVI file standard was not designed with this kind of functionality in mind. This function therefore only works well with standard ASCII characters. It will not work with non-European languages. Depending on the fonts used in the files, there may also be problems with accented characters or umlauts, and sometimes with ligatures.

Chapter 5

Embedding PostScript files into the DVI

The traditional way of using graphics with TeX does not include the graphics data directly in the DVI file. Instead, the DVI file contains only a link to a graphics file which resides on the hard disk. The advantage of this procedure is that the DVI file stays small, and that the graphics file can be modified independent of the document's TeX source. The method, however, becomes fairly inconvenient if you intend to archive the DVI file, or if you wish to send it to someone else: rather than handling a single file, you have to deal with a multitude of files, which need to be kept in exactly the place specified in the DVI file for everything to work.

For that reason, KDVI allows you to embed external PostScript® files into your DVI file. To embed all PostScript® files into a DVI file, use the menu entry Edit/Embed external PostScript files

WARNING

DVI files with embedded PostScript® work fine with most other DVI handling software, e.g. `xdvi`, `dvips` or `dvipdf`. One notable exception is the `dvipdfm` program, which currently ignores the embedded PostScript®. Since `dvipdfm` is used internally by the "Export to PDF" function of KDVI, expect problems when you use that function. The same issue shows with other software that uses embedded PostScript, e.g. the TeX `xy` macro package.

Chapter 6

Using inverse search

Inverse search is a very useful feature when you are writing a TeX document yourself. If everything is properly set up, you can click into KDVI's window with the middle mouse button (on some systems, when you don't have a three-button mouse, you can simultaneously use the left and the right button). After that, your favorite editor will open, load the TeX source file and jump to the proper paragraph. To use inverse search, do the following:

1. Produce a DVI file that contains inverse search information. This is explained in the section [Producing TeX files for inverse search](#) below. If you just want to test the inverse search feature, you can also use the example file `KDVI-features.dvi`
2. Let KDVI know which editor you would like to use. Choose an editor in the Preferences dialog (this dialog can be reached by choosing DVI Options in the Settings menu). The next section of this documentation, [Rendering Options](#), explains this dialog in more detail.
3. Some editors need to be started manually, or need additional configuration. You will find a description of all supported editors in the section [Setting up your editor for inverse search](#) below.
4. Test your setup. Open your DVI file in KDVI and use the middle mouse button to click into KDVI. The editor should pop up and display the TeX file.

6.1 Producing TeX files for inverse search

There are essentially two ways to produce DVI files which contain inverse search information: you can either use a TeX/LaTeX binary which generates and includes the necessary information automatically, or you can include an extra package which is written in TeX/LaTeX.

- A TeX binary which generates and includes the necessary information automatically is certainly the preferred method of including inverse search information. If you use version 2 or greater of the [TeX TeX distribution](#), you can use the 'src-specials' command line option of the `tex` or `latex` command, as follows.

```
tex --src-specials myfile.tex
```

or

```
latex --src-specials myfile.tex
```

- If you do not have a TeX binary which includes inverse search information natively, copy the files `srcltx.sty` and `srctex.sty` to the folder where your TeX file resides (you can do that by pressing the **Shift** key and left mouse button while the mouse pointer is on a hyperlink.) If you use LaTeX, add the line

```
\usepackage[active]{srcltx}
```

to the preamble of your LaTeX file. If you use plain TeX, the line

```
\include{srctex}
```

will do the trick.

TIP

While inverse search is extremely useful when you are typing a document yourself, it might be a good idea to remove the inverse search information before sending the DVI file to someone else.

6.2 Setting up your editor for inverse search

While inverse search works generally very well with most editors, some of them require a bit of extra care. This section explains how to configure your editor.

6.2.1 Emacs

Emacs works well with KDVI. The actual behavior of Emacs depends largely on the configuration. As usual, you can customize Emacs completely, if you are willing to fight your way through Lisp code.

KDVI uses the program **emacsclient** to remote control Emacs.

The KDVI Handbook

IMPORTANT

The program **emacsclient** requires that Emacs is running, and that the program Emacs Server is started inside Emacs. Inverse search will not work optimally unless you have started both Emacs and the Emacs Server.

To start the Emacs Server, you can do one of the following:

- In Emacs, start the Emacs Server by typing `M X server-start`
- Add the line

```
(server-start)
```

to your `.emacs` file. Restart Emacs

The KDVI Handbook

TIP

- Make sure that Emacs is installed. Try to start **emacs** from the command line.
- KDVI uses the command **emacsclient** to remote control Emacs. Make sure that **emacsclient** is available on the command line by trying the command **emacsclient** *Name of a text file*. This should open a new text in the Emacs editor.
- If **emacsclient** fails with an error message like `unable to connect to local`, make sure that Emacs is running. Furthermore, make sure that the Emacs Server is started by typing `M x server-start`.
- If you want the frame to be auto-raised, add the `raise-frame` function to `'server-switch-hook'` (do `M x customize-variable RET server-switch-hook` and enter the function name into the text field).
- If you have changed the buffer since your last save, Emacs will ask you: `Revert buffer from file ...? (yes or no)`. You will probably always want to say *no* here, since reverting means that the file is reread from disk, *causing all your changes since the last save to be lost!*

gnuclient's behavior of silently reloading the changed buffer is probably preferable — add the following lines to your `.emacs` file to emulate **gnuclient's** behavior with **emacsclient**:

```
(defadvice server-visit-files (around save-buffers last ↵
  activate)
  "Try to emulate gnuclient behavior with emacsclient.
  Works only for visiting one buffer at a time."
  (let* ((filen (car (car (ad-get-arg 0))))
         (buf (get-file-buffer filen))
         (this-buf-modified-p nil))
    ;;; the following is copied from server-visit-files ↵
    , with
    ;;; a modification for the 'verify-visited-file- ↵
    modtime' test
    (if (and buf (set-buffer buf))
        (if (file-exists-p filen)
            ;;; if the file has changed on disk, reload ↵
            it
            ;;; using 'find-file-noselect'
            (if (not (verify-visited-file-modtime buf))
                (progn
                  (find-file-noselect filen)
                  ;;; if user answered 'no', reset ↵
                  modtime anyway
                  ;;; so that server-visit-files doesn' ↵
                  t realize the
                  ;;; difference:
                  (set-visited-file-modtime)))
                ;;; if file exists no longer, we let server- ↵
                visit-files 10
                ;;; deal with that
                t)
            (setq buf (find-file-noselect filen)))
        (setq this-buf-modified-p (buffer-modified-p buf))
        (set-buffer buf)
        (set-buffer-modified-p nil)
        ad-do-it
        (set-buffer-modified-p this-buf-modified-p)))
```

6.2.2 Kate

KDE's editor Kate supports inverse search very well. No extra setup is required.

6.2.3 Kile

The LaTeX-editor system Kile, supports KDVI very well. No extra setup is necessary. Further information about Kile can be found at [Kile's homepage](#).

6.2.4 NEdit

NEdit generally works very well indeed. Clicking into the DVI file should open a new window. If the TeX file is already used in another window of NEdit, the newly opened window displays another view of the buffer. Otherwise, the TeX file is loaded. After opening the window, NEdit highlights the first line of the appropriate paragraph.

TIP

KDVI uses the command **ncl** to remote control NEdit. Make sure that **ncl** is available on the command line by trying the command **ncl -noask**. This should open an instance of the NEdit editor. If **ncl** is not available, you might be using an older version of NEdit. In that case, you should either upgrade to a more recent version, or you have to use the option User defined editor from the Options dialog.

6.2.5 XEmacs

XEmacs works well with KDVI. The actual behavior of XEmacs depends largely on the configuration. As usual, you can customize XEmacs completely, if you are willing to fight your way through Lisp code.

KDVI uses the program **gnuclient** to remote control XEmacs.

IMPORTANT

The program **gnuclient** requires that XEmacs is running, and that the program **gnuserv** is started inside XEmacs. Inverse search will not work unless you have started both XEmacs and **gnuserv**.

To start the **gnuserv** program, you can do one of the following:

- In XEmacs, start **gnuserv** by typing `M X gnuclient-start`
- Add the line

```
(gnuserv-start)
```

to your `.xemacs` file. If you use a more recent version of XEmacs, `.xemacs` will be a folder. In that case, you should append the line to the file `.xemacs-init.el`. Restart XEmacs

If you don't want to open a new frame for each editor call, and want the frame to be auto-raised, set 'Gnuserv Frame' to 'Use selected frame', and add the `raise-frame` function to 'Visit Hook'. Do `M x customize-group RET gnuserv` to make these settings.

TIP

- Make sure that XEmacs is installed. Try to start **xemacs** from the command line.
- KDVI uses the command `gnuserv` to remote control XEmacs. Make sure that **gnuclient** is available on the command line by trying the command **gnuclient** *Name of a text file*. This should open a new frame in the XEmacs editor.
- If `gnuserv` fails with an error message like `unable to connect to local`, make sure that XEmacs is running. Furthermore, make sure that `gnuserv` is started by typing `M X gnuserv-start`.
- If you don't want to open a new frame for each editor call, and want the frame to be auto-raised, set 'Gnuserv Frame' to 'Use selected frame', and add the 'raise-frame' function to 'Visit Hook'. Do `M+X customize-group RET gnuserv` to make these settings.

6.2.6 VI iMproved / GUI

The `gvim` variant of the `vi` editor supports inverse search very well. No extra setup is required.

Chapter 7

Forward search

The forward search functions allow you to jump from your editor directly into the associated position of the DVI file. Since forward search must be supported by your editor, only Emacs and XEmacs are currently supported. Other editors will hopefully join in soon.

To use forward search, you have to do the following:

- Set up your editor — this is described below.
- Add source file information to your DVI file, e.g. by using the package `srcltx`. This has been described in the section [Producing TeX files for inverse search](#).
- If you use Emacs and everything is properly set up, you just press `Ctrl+X Ctrl+J`, and KDVI pops up and jumps to the place which corresponds to the place of the TeX file which you are currently editing.

7.1 Setting up your editor for forward search

7.1.1 Emacs

In order to use forward search in Emacs, follow these steps:

- Download the following Emacs script, [kdvi-search.el](#) (press **Shift** and left mouse button the filename to download) and store it in a place where Emacs can access it — we recommend a folder `emacs-scripts`.
- Add the lines

```
(add-to-list 'load-path (expand-file-name "~/emacs-scripts" ↵  
  "/"))  
(require 'kdvi-search)
```

The KDVI Handbook

```
(add-hook 'LaTeX-mode-hook (lambda () (local-set-key "\C-x\ ←  
C-j" 'kdvi-jump-to-line)))  
(add-hook 'tex-mode-hook (lambda () (local-set-key "\C-x\C- ←  
j" 'kdvi-jump-to-line)))
```

to your `.emacs` file. Restart Emacs.

- Open Emacs, load a TeX file, produce the corresponding DVI file, and either enter the command `M+xkdvi-jump-to-line` or press **CtrlX CtrlJ**. It may happen that Emacs asks you for the name of a 'master file'. This is useful if you use a TeX file which includes other files: the master file is the top-level file which includes the others. Emacs will perhaps also ask to save the name of the master file 'as a local variable', i.e. as a comment at the very end of the file. Type either `yes` or `no` to continue.

TIP

- Make sure that Emacs is installed. Try to start **emacs** from the command line.
- If Emacs fails to start KDVI, you can find its output in the Buffer `kdvi-output`.

7.1.2 Kile

If you use Kile, no further setup is necessary.

7.1.3 XEmacs

To set up XEmacs, follow the steps for Emacs [above](#), but modify your `.xemacs` rather than your `.emacs` file. If you use a very recent version of XEmacs, `.xemacs` may be a folder. In that case, append the lines to `.xemacs/init.el`.

Chapter 8

The Preferences dialog

The Preferences dialog can be reached by choosing DVI Options in the Settings menu.

The dialog consists of two tabs, Fonts and Rendering.

8.1 Fonts Options

Traditionally, the TeX typesetter uses fonts that are generated by the **MetaFont** program. These fonts are stored in the PK format. While a carefully configured **MetaFont** system produces printouts of highest quality, its configuration requires serious expertise, **MetaFont** is not very good at producing fonts suitable for computer displays, and there are only few **MetaFont** fonts available for Asian languages.

To overcome these problems, newer TeX installations therefore include fonts that are stored in the "PostScript Type 1" format, which is a widely used font format in electronic publishing. KDVI is able to use both font formats.

The following picture shows the font options dialog of KDVI that can be used to control KDVI's use of the various font formats.

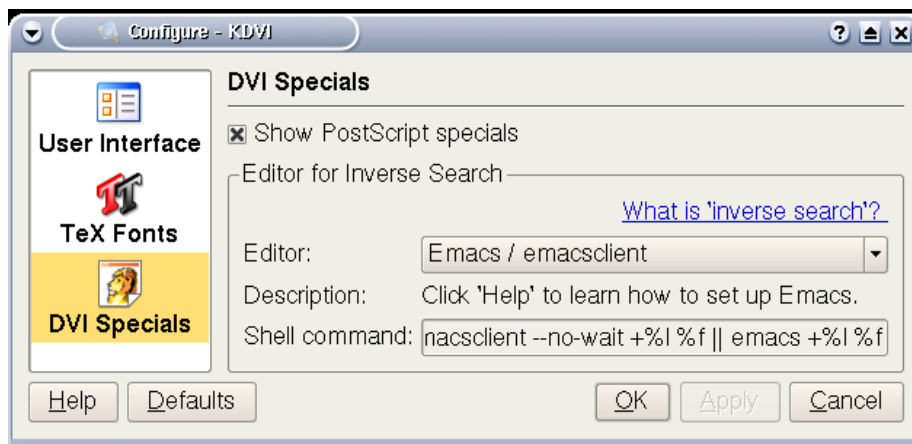


Use font hinting for Type 1 fonts, if available PostScript "Type 1" often contain "font hints", i.e. additional information that is supposed to help software produce better quality output on computer screens. The quality of the font hints varies from font to font, and you should experiment to see if enabling this option gives better results.

8.2 DVI specials Options

KDVI supports a large number of extensions to the original DVI format, e.g. hyperlinks, graphic file inclusion or embedded source file information. These extensions are known as "DVI specials". A full account of specials supported by KDVI can be found in [this document](#).

The DVI specials dialog help you to configure support for some specials.



Show PostScript specials If this option is checked, KDVI will display PostScript® graphics which are embedded into the DVI file. You probably want to set this option.

If an external PostScript® file could not be found, KDVI will draw a red warning box in its place. Unfortunately, rendering PostScript® graphics is very slow in the current version of KDVI. We will improve on the speed in later versions. If this option is off, KDVI will either draw a gray box as a placeholder for the graphics, or it will leave the space blank.

NOTE

There is no standard way to embed PostScript® graphics into a DVI file. It may therefore happen that KDVI cannot properly display a graphic which works fine with other programs. Older versions of **xdvi** and **dvips** support the execution of external commands. This is a bad security risk and therefore deliberately not implemented in KDVI. Technical information about supported ways to include PostScript® can be found in the document `KDVI-feature-s.dvi`.

Editor for inverse search If you intend to use [inverse search](#), a very useful feature if you write TeX documents yourself, you have to specify which editor you are going to use, and how this editor can be started by KDVI. In the example shown, the user has opted for the NEdit editor. If you use one of the pre-configured editors from the Editor combobox, then you don't have to do anything else. If you wish to use a different editor, chose User-defined editor from the Editor combobox and enter the command line which will be used to start your editor. Use the placeholders `%f` and `%l` which will be replaced with the name of the TeX file, and the line of the TeX file, respectively.

If you use an editor which is not supported, please send us an email at kebekus@kde.org and tell us about the command line you use and how you have configured your editor.

Chapter 9

Frequently asked questions

1. *What happens when KDVI displays the message KDVI is currently generating bitmap fonts, and why does the procedure take so long?*

Many of the fonts which are typically used in a TeX document must be generated by the MetaFont system. Metafont is a language similar to TeX (included in most TeX distributions) which takes a description of the font outline, and produces a rasterized version (.pk file) of the font which can then be send to a printer or be used in a previewing program like KDVI. Metafont goes out of its way to produce the best possible output for your printer. For instance, it knows that a pixel of an inkjet printer is a roundish blot, and that nearby pixels tend to smear into each other. In contrast, a pixel on a laser printer is rectangular, but an isolated pixel is very often not rendered at all. Generating such highly optimized bitmap fonts is naturally rather time-consuming, in particular since typical TeX documents use a large number of different fonts. We can only ask for your patience. To ease the matter somewhat, most distributions of TeX store the .pk files for a limited time, e.g. 100 days. Therefore, if you access the same document more than once, the .pk files will be reused.

2. *What is a MetaFont Mode?*

In order to produce bitmap fonts which are optimized for your printer (see the answer to the first question), Metafont comes with a database of printing engines — look for a file called `modes.mf`. A Metafont Mode is just the name of a database entry. For example, the name 'ljfour' refers to the entry in the database that describes a Hewlett-Packard® LaserJet 4 printer. A MetaFont Mode is usually followed by a number, the resolution. The LaserJet, for instance, can print in both 300 and 600 dots per inch. Thus, 'ljfour/600' would be a full description.

Chapter 10

Credits and Licenses

KDVI

KDVI is based on based on the stand-alone-program KDVI 0.4.3 by Markku Hihnala. That program is in turn based on xdvi version 18f which has many authors.

Documentation is copyright 2001-2004, Stefan Kebekus kebekus@kde.org

This program is licensed under the terms of the [GNU General Public License](#).

This documentation is licensed under the terms of the [GNU Free Documentation License](#).