Hypertext marks in LaTeX: a manual for **hyperref**

Sebastian Rahtz   Heiko Oberdiek

April 2012

Contents

1 Introduction

The package derives from, and builds on, the work of the HyperTeX project, described at http://xxx.lanl.gov/hypertex/. It extends the functionality of all the LaTeX cross-referencing commands (including the table of contents, bibliographies etc) to produce `\special` commands which a driver

can turn into hypertext links; it also provides new commands to allow the user to write *ad hoc* hypertext links, including those to external documents and URLs.

This manual provides a brief overview of the hyperref package. For more details, you should read the additional documentation distributed with the package, as well as the complete documentation by processing hyperref.dtx. You should also read the chapter on hyperref in *The LATEX Web Companion*, where you will find additional examples.

The HyperTEX specification[1] says that conformant viewers/translators must recognize the following set of \special constructs:

href: html:

name: html:

end: html:

image: html:<img src = "href_string">

base_name: html:<base href = "href_string">

The *href*, *name* and *end* commands are used to do the basic hypertext operations of establishing links between sections of documents. The *image* command is intended (as with current HTML viewers) to place an image of arbitrary graphical format on the page in the current location. The *base_name* command is be used to communicate to the DVI viewer the full (URL) location of the current document so that files specified by relative URLs may be retrieved correctly.

The *href* and *name* commands must be paired with an *end* command later in the TEX file—the TEX commands between the two ends of a pair form an *anchor* in the document. In the case of an *href* command, the *anchor* is to be highlighted in the *DVI viewer*, and when clicked on will cause the scene to shift to the destination specified by *href_string*. The *anchor* associated with a name command represents a possible location to which other hypertext links may refer, either as local references (of the form href="#name_string" with the *name_string* identical to the one in the name command) or as part of a URL (of the form *URL#name_string*). Here *href_string* is a valid URL or local identifier, while *name_string* could be any string at all: the only caveat is that '"' characters should be escaped with a backslash (\), and if it looks like a URL name it may cause problems.

However, the drivers intended to produce *only* PDF use literal PostScript or PDF \special commands. The commands are defined in configuration files for different drivers, selected by package options; at present, the following drivers are supported:

hypertex DVI processors conforming to the HyperTEX guidelines (i.e. xdvi, dvips (with the -z option), OzTEX, and Textures)

dvips produces \special commands tailored for dvips

dvipsone produces \special commands tailored for dvipsone

ps2pdf a special case of output suitable for processing by earlier versions of Ghostscript's PDF writer; this is basically the same as that for dvips, but a few variations remained before version 5.21

tex4ht produces \special commands for use with TEX4ht

pdftex pdfTEX, Hàn Thê Thành's TEX variant that writes PDF directly

dvipdfm produces \special commands for Mark Wicks' DVI to PDF driver dvipdfm

---

[1]This is borrowed from an article by Arthur Smith.

dvipdfmx produces \special commands for driver dvipdfmx, a successor of dvipdfm

dviwindo produces \special commands that Y&Y's Windows previewer interprets as hypertext jumps within the previewer

vtex produces \special commands that MicroPress' HTML and PDF-producing TeX variants interpret as hypertext jumps within the previewer

textures produces \special commands that Textures interprets as hypertext jumps within the previewer

xetex produces \special commands for XeTeX

Output from dvips or dvipsone must be processed using Acrobat Distiller to obtain a PDF file.[2] The result is generally preferable to that produced by using the hypertex driver, and then processing with dvips -z, but the DVI file is not portable. The main advantage of using the HyperTeX \special commands is that you can also use the document in hypertext DVI viewers, such as xdvi.

driverfallback If a driver is not given and cannot be autodetected, then use the driver option, given as value to this option driverfallback. Example:

    driverfallback=dvipdfm

Autodetected drivers (pdftex, xetex, vtex, vtexpdfmark) are recognized from within TeX and therefore cannot be given as value to option driverfallback. However a DVI driver program is run after the TeX run is finished. Thus it cannot be detected at TeX macro level. Then package hyperref uses the driver, given by driverfallback. If the driver is already specified or can be autodetected, then option driverfallback is ignored.

2 Implicit behavior

This package can be used with more or less any normal LaTeX document by specifying in the document preamble

\usepackage{hyperref}

Make sure it comes *last* of your loaded packages, to give it a fighting chance of not being over-written, since its job is to redefine many LaTeX commands. Hopefully you will find that all cross-references work correctly as hypertext. For example, \section commands will produce a bookmark and a link, whereas \section* commands will only show links when paired with a corresponding \addcontentsline command.

In addition, the hyperindex option (see below) attempts to make items in the index by hyperlinked back to the text, and the option backref inserts extra 'back' links into the bibliography for each entry. Other options control the appearance of links, and give extra control over PDF output. For example, colorlinks, as its name well implies, colors the links instead of using boxes; this is the option used in this document.

---

[2]Make sure you turn off the partial font downloading supported by dvips and dvipsone in favor of Distiller's own system.

3 Package options

All user-configurable aspects of hyperref are set using a single 'key=value' scheme (using the keyval package) with the key Hyp. The options can be set either in the optional argument to the \usepackage command, or using the \hypersetup macro. When the package is loaded, a file hyperref.cfg is read if it can be found, and this is a convenient place to set options on a site-wide basis.

As an example, the behavior of a particular file could be controlled by:

- a site-wide hyperref.cfg setting up the look of links, adding backreferencing, and setting a PDF display default:

  \hypersetup{backref,
  pdfpagemode=FullScreen,
  colorlinks=true}

- A global option in the file, which is passed down to hyperref:

  \documentclass[dvips]{article}

- File-specific options in the \usepackage commands, which override the ones set in hyperref.cfg:

  \usepackage[colorlinks=false]{hyperref}
  \hypersetup{pdftitle={A Perfect Day}}

As seen in the previous example, information entries (pdftitle, pdfauthor, …) should be set after the package is loaded. Otherwise LaTeX expands the values of these options prematurely. Also LaTeX strips spaces in options. Especially option 'pdfborder' requires some care. Curly braces protect the value, if given as package option. They are not necessary in \hypersetup.

\usepackage[pdfborder={0 0 0}]{hyperref}
\hypersetup{pdfborder=0 0 0}

Package 'kvoptions-patch' patches LaTeX to make it aware of key value options and to prevent premature value expansions.

Some options can be given at any time, but many are restricted: before \begin{document}, only in \usepackage[...]{hyperref}, before first use, etc.

In the key descriptions that follow, many options do not need a value, as they default to the value true if used. These are the ones classed as 'boolean'. The values true and false can always be specified, however.

3.1 General options

Firstly, the options to specify general behavior and page size.

draft	boolean	*false*	all hypertext options are turned off
final	boolean	*true*	all hypertext options are turned on
debug	boolean	*false*	extra diagnostic messages are printed in the log file
verbose	boolean	*false*	same as debug
implicit	boolean	*true*	redefines LaTeX internals
hypertexnames	boolean	*true*	use guessable names for links
naturalnames	boolean	*false*	use LaTeX-computed names for links
setpagesize	boolean	*true*	sets page size by special driver commands

3.2 Configuration options

raiselinks	boolean	*true*	In the hypertex driver, the height of links is normally calculated by the driver as simply the base line of contained text; this options forces \special commands to reflect the real height of the link (which could contain a graphic)
breaklinks	boolean	*false*	Allows link text to break across lines; since this cannot be accommodated in PDF, it is only set true by default if the pdftex driver is used. This makes links on multiple lines into different PDF links to the same target.
pageanchor	boolean	*true*	Determines whether every page is given an implicit anchor at the top left corner. If this is turned off, \printindex will not contain valid hyperlinks.
plainpages	boolean	*false*	Forces page anchors to be named by the Arabic form of the page number, rather than the formatted form.
nesting	boolean	*false*	Allows links to be nested; no drivers currently support this.

Note for option breaklinks: The correct value is automatically set according to the driver features. It can be overwritten for drivers that do not support broken links. However, at any case, the link area will be wrong and displaced.

3.3 Backend drivers

If no driver is specified, the package tries to find a driver in the following order:

1. Autodetection, some TeX processors can be detected at TeX macro level (pdfTeX, XeTeX, VTeX).

2. Option driverfallback. If this option is set, its value is taken as driver option.

3. Macro \Hy@defaultdriver. The macro takes a driver file name (without file extension).

4. Package default is hypertex.

Many distributions are using a driver file hypertex.cfg that define \Hy@defaultdriver with hdvips. This is recommended because driver dvips provides much more features than hypertex for PDF generation.

driverfallback	Its value is used as driver option if the driver is not given or autodetected.
dvipdfm	Sets up hyperref for use with the dvipdfm driver.
dvipdfmx	Sets up hyperref for use with the dvipdfmx driver.
dvips	Sets up hyperref for use with the dvips driver.
dvipsone	Sets up hyperref for use with the dvipsone driver.
dviwindo	Sets up hyperref for use with the dviwindo Windows previewer.
hypertex	Sets up hyperref for use with the HyperTeX-compliant drivers.
latex2html	Redefines a few macros for compatibility with latex2html.

nativepdf	An alias for dvips
pdfmark	An alias for dvips
pdftex	Sets up hyperref for use with the pdftex program.
ps2pdf	Redefines a few macros for compatibility with Ghostscript's PDF writer, otherwise identical to dvips.
tex4ht	For use with TeX4ht
textures	For use with Textures
vtex	For use with MicroPress' VTeX; the PDF and HTML backends are detected automatically.
vtexpdfmark	For use with VTeX's PostScript backend.
xetex	For use with XeTeX(using backend for dvipdfm).

If you use dviwindo, you may need to redefine the macro `\wwwbrowser` (the default is C:\netscape\netscape) to tell dviwindo what program to launch. Thus, users of Internet Explorer might add something like this to hyperref.cfg:

```
\renewcommand{\wwwbrowser}{C:\string\Program\space
 Files\string\Plus!\string\Microsoft\space
 Internet\string\iexplore.exe}
```

3.4 Extension options

extension	text		Set the file extension (e.g. dvi) which will be appended to file links created if you use the xr package.
hyperfigures	boolean		
backref	text	*false*	Adds 'backlink' text to the end of each item in the bibliography, as a list of section numbers. This can only work properly *if* there is a blank line after each `\bibitem`. Supported values are **section**, **slide**, **page**, **none**, or **false**. If no value is given, **section** is taken as default.
pagebackref	boolean	*false*	Adds 'backlink' text to the end of each item in the bibliography, as a list of page numbers.
hyperindex	boolean	*true*	Makes the page numbers of index entries into hyperlinks. Relays on unique page anchors (**pageanchor**, ...)
pageanchors and plainpages=false.			
hyperfootnotes	boolean	*true*	Makes the footnote marks into hyperlinks to the footnote text. Easily broken ...
encap			Sets encap character for hyperindex
linktocpage	boolean	*false*	make page number, not text, be link on TOC, LOF and LOT
breaklinks	boolean	*false*	allow links to break over lines by making links over multiple lines into PDF links to the same target

`colorlinks`	boolean	*false*	Colors the text of links and anchors. The colors chosen depend on the the type of link. At present the only types of link distinguished are citations, page references, URLs, local file references, and other links. In spite of colored boxes, the colored text remains when printing.
`linkcolor`	color	*red*	Color for normal internal links.
`anchorcolor`	color	*black*	Color for anchor text.
`citecolor`	color	*green*	Color for bibliographical citations in text.
`filecolor`	color	*cyan*	Color for URLs which open local files.
`menucolor`	color	*red*	Color for Acrobat menu items.
`runcolor`	color	*filecolor*	Color for run links (launch annotations).
`urlcolor`	color	*magenta*	Color for linked URLs.
`allcolors`	color		Set all color options (without border and field options).
`frenchlinks`	boolean	*false*	Use small caps instead of color for links.
`hidelinks`			Hide links (removing color and border).

Note that all color names must be defined before use, following the normal system of the standard LATEX color package.

3.5 PDF-specific display options

`bookmarks`	boolean	*true*	A set of Acrobat bookmarks are written, in a manner similar to the table of contents, requiring two passes of LATEX. Some postprocessing of the bookmark file (file extension .out) may be needed to translate LATEX codes, since bookmarks must be written in PDFEncoding. To aid this process, the .out file is not rewritten by LATEX if it is edited to contain a line \let\WriteBookmarks\relax
`bookmarksopen`	boolean	*false*	If Acrobat bookmarks are requested, show them with all the subtrees expanded.
`bookmarksopenlevel`	parameter		level (\maxdimen) to which bookmarks are open
`bookmarksnumbered`	boolean	*false*	If Acrobat bookmarks are requested, include section numbers.
`bookmarkstype`	text	*toc*	to specify which 'toc' file to mimic

`CJKbookmarks`	boolean	*false*	This option should be used to produce CJK bookmarks. Package `hyperref` supports both normal and preprocessed mode of the CJK package; during the creation of bookmarks, it simply replaces CJK's macros with special versions which expand to the corresponding character codes. Note that without the 'unicode' option of hyperref you get PDF files which actually violate the PDF specification because non-Unicode character codes are used – some PDF readers localized for CJK languages (most notably Acroread itself) support this. Also note that option 'CJKbookmarks' cannot be used together with option 'unicode'.
			No mechanism is provided to translate non-Unicode bookmarks to Unicode; for portable PDF documents only Unicode encoding should be used.
`pdfhighlight`	name	*/I*	How link buttons behave when selected; /I is for inverse (the default); the other possibilities are /N (no effect), /O (outline), and /P (inset highlighting).
`citebordercolor`	RGB color	*0 1 0*	The color of the box around citations
`filebordercolor`	RGB color	*0 .5 .5*	The color of the box around links to files
`linkbordercolor`	RGB color	*1 0 0*	The color of the box around normal links
`menubordercolor`	RGB color	*1 0 0*	The color of the box around Acrobat menu links
`urlbordercolor`	RGB color	*0 1 1*	The color of the box around links to URLs
`runbordercolor`	RGB color	*0 .7 .7*	Color of border around 'run' links
`allbordercolors`			Set all border color options
`pdfborder`		*0 0 1*	The style of box around links; defaults to a box with lines of 1pt thickness, but the colorlinks option resets it to produce no border.

Note that the color of link borders can be specified *only* as 3 numbers in the range 0..1, giving an RGB color. You cannot use colors defined in TEX. Since version 6.76a this is no longer true. Especially with the help of package xcolor the usual color specifications of package (x)color can be used. For further information see description of package hycolor.

The bookmark commands are stored in a file called *jobname*.out. The files is not processed by LATEX so any markup is passed through. You can postprocess this file as needed; as an aid for this, the .out file is not overwritten on the next TEX run if it is edited to contain the line

`\let\WriteBookmarks\relax`

3.6 PDF display and information options

`baseurl`		URL	Sets the base URL of the PDF document

pdfpagemode	text	*empty*	Determines how the file is opening in Acrobat; the possibilities are `UseNone`, `UseThumbs` (show thumbnails), `UseOutlines` (show bookmarks), `FullScreen`, `UseOC` (PDF 1.5), and `UseAttachments` (PDF 1.6). If no mode if explicitly chosen, but the bookmarks option is set, `UseOutlines` is used.
pdftitle	text		Sets the document information Title field
pdfauthor	text		Sets the document information Author field
pdfsubject	text		Sets the document information Subject field
pdfcreator	text		Sets the document information Creator field
pdfproducer	text		Sets the document information Producer field
pdfkeywords	text		Sets the document information Keywords field
pdftrapped	text	*empty*	Sets the document information Trapped entry. Possible values are `True`, `False` and `Unknown`. An empty value means, the entry is not set.
pdfinfo	key value list	*empty*	Alternative interface for setting the document information.
pdfview	text	*XYZ*	Sets the default PDF 'view' for each link
pdfstartpage	text	*1*	Determines on which page the PDF file is opened.
pdfstartview	text	*Fit*	Set the startup page view
pdfremotestartview	text	*Fit*	Set the startup page view of remote PDF files
pdfpagescrop	n n n n		Sets the default PDF crop box for pages. This should be a set of four numbers
pdfcenterwindow	boolean	*false*	position the document window in the center of the screen
pdfdirection	text	*empty*	direction setting
pdfdisplaydoctitle	boolean	*false*	display document title instead of file name in title bar
pdfduplex	text	*empty*	paper handling option for print dialog
pdffitwindow	boolean	*false*	resize document window to fit document size
pdflang	text	*empty*	PDF language identifier (RFC 3066)
pdfmenubar	boolean	*true*	make PDF viewer's menu bar visible
pdfnewwindow	boolean	*false*	make links that open another PDF file start a new window
pdfnonfullscreenpagemode	boolean	*empty*	page mode setting on exiting full-screen mode
pdfnumcopies	integer	*empty*	number of printed copies
pdfpagelayout	text	*empty*	set layout of PDF pages
pdfpagelabels	boolean	*true*	set PDF page labels
pdfpagetransition	text	*empty*	set PDF page transition style
pdfpicktraybypdfsize	text	*empty*	set option for print dialog
pdfprintarea	text	*empty*	set /PrintArea of viewer preferences
pdfprintclip	text	*empty*	set /PrintClip of viewer preferences
pdfprintpagerange	n n (n n)*	*empty*	set /PrintPageRange of viewer preferences
pdfprintscaling	text	*empty*	page scaling option for print dialog (option /PrintScaling of viewer preferences, PDF 1.6); valid values are `None` and `AppDefault`
pdftoolbar	boolean	*true*	make PDF toolbar visible
pdfviewarea	text	*empty*	set /ViewArea of viewer preferences
pdfviewclip	text	*empty*	set /ViewClip of viewer preferences

`pdfwindowui`	boolean	*true*	make PDF user interface elements visible
`unicode`	boolean	*false*	Unicode encoded PDF strings

Each link in Acrobat carries its own magnification level, which is set using PDF coordinate space, which is not the same as TeX's. The unit is bp and the origin is in the lower left corner. See also `\hypercalcbp` that is explained on page 18. pdfTeX works by supplying default values for `XYZ` (horizontal × vertical × zoom) and `FitBH`. However, drivers using `pdfmark` do not supply defaults, so `hyperref` passes in a value of -32768, which causes Acrobat to set (usually) sensible defaults. The following are possible values for the `pdfview`, `pdfstartview` and `pdfremotestartview` parameters.

`XYZ`	*left top zoom*	Sets a coordinate and a zoom factor. If any one is null, the source link value is used. *null null null* will give the same values as the current page.
`Fit`		Fits the page to the window.
`FitH`	*top*	Fits the width of the page to the window.
`FitV`	*left*	Fits the height of the page to the window.
`FitR`	*left bottom right top*	Fits the rectangle specified by the four coordinates to the window.
`FitB`		Fits the page bounding box to the window.
`FitBH`	*top*	Fits the width of the page bounding box to the window.
`FitBV`	*left*	Fits the height of the page bounding box to the window.

The `pdfpagelayout` can be one of the following values.

`SinglePage`	Displays a single page; advancing flips the page
`OneColumn`	Displays the document in one column; continuous scrolling.
`TwoColumnLeft`	Displays the document in two columns, odd-numbered pages to the left.
`TwoColumnRight`	Displays the document in two columns, odd-numbered pages to the right.
`TwoPageLeft`	Displays two pages, odd-numbered pages to the left (since PDF 1.5).
`TwoPageRight`	Displays two pages, odd-numbered pages to the right (since PDF 1.5).

Finally, the `pdfpagetransition` can be one of the following values, where */Di* stands for direction of motion in degrees, generally in 90° steps, */Dm* is a horizontal (`/H`) or vertical (`/V`) dimension (e.g. `Blinds /Dm /V`), and */M* is for motion, either in (`/I`) or out (`/O`).

`Blinds`	/Dm	Multiple lines distributed evenly across the screen sweep in the same direction to reveal the new page.
`Box`	/M	A box sweeps in or out.
`Dissolve`		The page image dissolves in a piecemeal fashion to reveal the new page.
`Glitter`	/Di	Similar to Dissolve, except the effect sweeps across the screen.
`Split`	/Dm /M	Two lines sweep across the screen to reveal the new page.

Wipe	/Di	A single line sweeps across the screen to reveal the new page.

3.7 Option `pdfinfo`

The information entries can be set using `pdftitle`, `pdfsubject`, ... Option `pdfinfo` provides an alternative interface. It takes a key value list. The key names are the names that appear in the PDF information dictionary directly. Known keys such as `Title`, `Subject`, `Trapped` and other are mapped to options `pdftitle`, `subject`, `trapped`, ...Unknown keys are added to the information dictionary. Their values are text strings (see PDF specification). Example:

```
\hypersetup{
  pdfinfo={
    Title={My Title},
    Subject={My Subject},
    NewKey={Foobar},
    % ...
  }
}
```

3.8 Big alphabetical list

The following is a complete listing of available options for hyperref, arranged alphabetically.

anchorcolor	*black*	set color of anchors
backref	*false*	do bibliographical back references
baseurl	*empty*	set base URL for document
bookmarks	*true*	make bookmarks
bookmarksnumbered	*false*	put section numbers in bookmarks
bookmarksopen	*false*	open up bookmark tree
bookmarksopenlevel	\maxdimen	level to which bookmarks are open
bookmarkstype	*toc*	to specify which 'toc' file to mimic
breaklinks	*false*	allow links to break over lines
CJKbookmarks	*false*	to produce CJK bookmarks
citebordercolor	*0 1 0*	color of border around cites
citecolor	*green*	color of citation links
colorlinks	*false*	color links
	true	(tex4ht, dviwindo)
debug	*false*	provide details of anchors defined; same as verbose
draft	*false*	do not do any hyperlinking
dvipdfm		use dvipdfm backend
dvipdfmx		use dvipdfmx backend
dvips		use dvips backend
dvipsone		use dvipsone backend
dviwindo		use dviwindo backend
encap		to set encap character for hyperindex
extension	*dvi*	suffix of linked files
filebordercolor	*0 .5 .5*	color of border around file links
filecolor	*cyan*	color of file links
final	*true*	opposite of option draft
frenchlinks	*false*	use small caps instead of color for links

`hyperfigures`	*false*	make figures hyper links
`hyperfootnotes`	*true*	set up hyperlinked footnotes
`hyperindex`	*true*	set up hyperlinked indices
`hypertex`		use HyperTeX backend
`hypertexnames`	*true*	use guessable names for links
`implicit`	*true*	redefine LaTeX internals
`latex2html`		use LaTeX2HTML backend
`linkbordercolor`	*1 0 0*	color of border around links
`linkcolor`	*red*	color of links
`linktocpage`	*false*	make page number, not text, be link on TOC, LOF and LOT
`menubordercolor`	*1 0 0*	color of border around menu links
`menucolor`	*red*	color for menu links
`nativepdf`	*false*	an alias for dvips
`naturalnames`	*false*	use LaTeX-computed names for links
`nesting`	*false*	allow nesting of links
`pageanchor`	*true*	put an anchor on every page
`pagebackref`	*false*	backreference by page number
`pdfauthor`	*empty*	text for PDF Author field
`pdfborder`	*0 0 1*	width of PDF link border
	0 0 0	(colorlinks)
`pdfcenterwindow`	*false*	position the document window in the center of the screen
`pdfcreator`	*LaTeX with hyperref package*	text for PDF Creator field
`pdfdirection`	*empty*	direction setting
`pdfdisplaydoctitle`	*false*	display document title instead of file name in title bar
`pdfduplex`	*empty*	paper handling option for print dialog
`pdffitwindow`	*false*	resize document window to fit document size
`pdfhighlight`	*/I*	set highlighting of PDF links
`pdfinfo`	*empty*	alternative interface for setting document information
`pdfkeywords`	*empty*	text for PDF Keywords field
`pdflang`	*empty*	PDF language identifier (RFC 3066)
`pdfmark`	*false*	an alias for dvips
`pdfmenubar`	*true*	make PDF viewer's menu bar visible
`pdfnewwindow`	*false*	make links that open another PDF file start a new window
`pdfnonfullscreenpagemode`	*empty*	page mode setting on exiting full-screen mode
`pdfnumcopies`	*empty*	number of printed copies
`pdfpagelayout`	*empty*	set layout of PDF pages
`pdfpagemode`	*empty*	set default mode of PDF display
`pdfpagelabels`	*true*	set PDF page labels
`pdfpagescrop`	*empty*	set crop size of PDF document
`pdfpagetransition`	*empty*	set PDF page transition style
`pdfpicktraybypdfsize`	*empty*	set option for print dialog
`pdfprintarea`	*empty*	set /PrintArea of viewer preferences
`pdfprintclip`	*empty*	set /PrintClip of viewer preferences
`pdfprintpagerange`	*empty*	set /PrintPageRange of viewer preferences
`pdfprintscaling`	*empty*	page scaling option for print dialog

pdfproducer	*empty*	text for PDF Producer field
pdfremotestartview	*Fit*	starting view of remote PDF documents
pdfstartpage	*1*	page at which PDF document opens
pdfstartview	*Fit*	starting view of PDF document
pdfsubject	*empty*	text for PDF Subject field
pdftex		use pdfTeX backend
pdftitle	*empty*	text for PDF Title field
pdftoolbar	*true*	make PDF toolbar visible
pdftrapped	*empty*	Sets the document information Trapped entry. Possible values are True, False and Unknown. An empty value means, the entry is not set.
pdfview	*XYZ*	PDF 'view' when on link traversal
pdfviewarea	*empty*	set /ViewArea of viewer preferences
pdfviewclip	*empty*	set /ViewClip of viewer preferences
pdfwindowui	*true*	make PDF user interface elements visible
plainpages	*false*	do page number anchors as plain Arabic
ps2pdf		use ps2pdf backend
raiselinks	*false*	raise up links (for HyperTeX backend)
runbordercolor	*0 .7 .7*	color of border around 'run' links
runcolor	*filecolor*	color of 'run' links
setpagesize	*true*	set page size by special driver commands
tex4ht		use TeX4ht backend
textures		use Textures backend
unicode	*false*	Unicode encoded pdf strings
urlbordercolor	*0 1 1*	color of border around URL links
urlcolor	*magenta*	color of URL links
verbose	*false*	be chatty
vtex		use VTeX backend
xetex		use XeTeX backend

4 Additional user macros

If you need to make references to URLs, or write explicit links, the following low-level user macros are provided:

> \href[*options*]{*URL*}{*text*}

The *text* is made a hyperlink to the *URL*; this must be a full URL (relative to the base URL, if that is defined). The special characters # and ˜ do *not* need to be escaped in any way.

The optional argument *options* recognizes the hyperref options `pdfremotestartview`, `pdfnewwindow` and the following key value options:

page: Specifies the start page number of remote PDF documents. First page is 1.

ismap: Boolean key, if set to |true|, the URL should appended by the coordinates as query parameters by the PDF viewer.

nextactionraw: The value of key |/Next| of action dictionaries, see PDF specification.

> \url{*URL*}

Similar to \href{*URL*}{\nolinkurl{*URL*}}. Depending on the driver \href also tries to detect the link type. Thus the result can be a url link, file link, …

> \nolinkurl{*URL*}

Write *URL* in the same way as \url, without creating a hyperlink.

> \hyperbaseurl{*URL*}

A base *URL* is established, which is prepended to other specified URLs, to make it easier to write portable documents.

> \hyperimage{*imageURL*}{*text*}

The link to the image referenced by the URL is inserted, using *text* as the anchor.

For drivers that produce HTML, the image itself is inserted by the browser, with the *text* being ignored completely.

> \hyperdef{*category*}{*name*}{*text*}

A target area of the document (the *text*) is marked, and given the name *category.name*

> \hyperref{*URL*}{*category*}{*name*}{*text*}

text is made into a link to *URL#category.name*

> \hyperref[*label*]{*text*}

text is made into a link to the same place as \ref{*label*} would be linked.

> \hyperlink{*name*}{*text*}

> \hypertarget{*name*}{*text*}

A simple internal link is created with \hypertarget, with two parameters of an anchor *name*, and anchor *text*. \hyperlink has two arguments, the name of a hypertext object defined somewhere by \hypertarget, and the *text* which be used as the link on the page.

Note that in HTML parlance, the \hyperlink command inserts a notional # in front of each link, making it relative to the current testdocument; \href expects a full URL.

> \phantomsection

This sets an anchor at this location. It works similar to \hypertarget{}{} with an automatically chosen anchor name. Often it is used in conjunction with \addcontentsline for sectionlike things (index, bibliography, preface). \addcontentsline refers to the latest previous location where an anchor is set. Example:

```
\cleardoublepage
\phantomsection
\addcontentsline{toc}{chapter}{\indexname}
\printindex
```

Now the entry in the table of contents (and bookmarks) for the index points to the start of the index page, not to a location before this page.

`\autoref{`*label*`}`

This is a replacement for the usual `\ref` command that places a contextual label in front of the reference. This gives your users a bigger target to click for hyperlinks (e.g. 'section 2' instead of merely the number '2').

The label is worked out from the context of the original `\label` command by hyperref by using the macros listed below (shown with their default values). The macros can be (re)defined in documents using `\(re)newcommand`; note that some of these macros are already defined in the standard document classes. The mixture of lowercase and uppercase initial letters is deliberate and corresponds to the author's practice.

For each macro below, hyperref checks `\*autorefname` before `\*name`. For instance, it looks for `\figureautorefname` before `\figurename`.

Macro	*Default*
`\figurename`	Figure
`\tablename`	Table
`\partname`	Part
`\appendixname`	Appendix
`\equationname`	Equation
`\Itemname`	item
`\chaptername`	chapter
`\sectionname`	section
`\subsectionname`	subsection
`\subsubsectionname`	subsubsection
`\paragraphname`	paragraph
`\Hfootnotename`	footnote
`\AMSname`	Equation
`\theoremname`	Theorem
`\page`	page

Example for a redefinition if babel is used:

```
\usepackage[ngerman]{babel}
\addto\extrasngerman{%
  \def\subsectionautorefname{Unterkapitel}%
}
```

Hint: `\autoref` works via the counter name that the reference is based on. Sometimes `\autoref` chooses the wrong name, if the counter is used for different things. For example, it happens with `\newtheorem` if a lemma shares a counter with theorems. Then package aliascnt provides a method to generate a simulated second counter that allows the differentiation between theorems and lemmas:

```
\documentclass{article}

\usepackage{aliascnt}
\usepackage{hyperref}

\newtheorem{theorem}{Theorem}
```

```
\newaliascnt{lemma}{theorem}
\newtheorem{lemma}[lemma]{Lemma}
\aliascntresetthe{lemma}

\providecommand*{\lemmaautorefname}{Lemma}

\begin{document}

We will use \autoref{a} to prove \autoref{b}.

\begin{lemma}\label{a}
  Nobody knows.
\end{lemma}

\begin{theorem}\label{b}
  Nobody is right.
\end{theorem}.

\end{document}
```

`\autopageref{`*label*`}`

It replaces `\pageref` and adds the name for page in front of the page reference. First `\pageautorefname` is checked before `\pagename`.

For instances where you want a reference to use the correct counter, but not to create a link, there are starred forms:

`\ref*{`*label*`}`

`\pageref*{`*label*`}`

`\autoref*{`*label*`}`

`\autopageref*{`*label*`}`

A typical use would be to write

`\hyperref[other]{that nice section (\ref*{other}) we read before}`

We want `\ref*{other}` to generate the correct number, but not to form a link, since we do this ourselves with `\hyperref`.

`\pdfstringdef{`*macroname*`}{`T_EX*string*`}`

`\pdfstringdef` returns a macro containing the PDF string. (Currently this is done globally, but do not rely on it.) All the following tasks, definitions and redefinitions are made in a group to keep them local:

- Switching to PD1 or PU encoding

- Defining the "octal sequence commands" (\345): \edef\3{\string\3}

- Special glyphs of TeX: \{, \%, \&, \space, \dots, etc.

- National glyphs (german.sty, french.sty, etc.)

- Logos: \TeX, \eTeX, \MF, etc.

- Disabling commands that do not provide useful functionality in bookmarks: \label, \index, \glossary, \discretionary, \def, \let, etc.

- LaTeX's font commands like \textbf, etc.

- Support for \xspace provided by the xspace package

In addition, parentheses are protected to avoid the danger of unsafe unbalanced parentheses in the PDF string. For further details, see Heiko Oberdiek's EuroTeX paper distributed with hyperref.

4.1 Bookmark macros

4.1.1 Setting bookmarks

Usually hyperref automatically adds bookmarks for \section and similar macros. But they can also set manually.

    \pdfbookmark[*level*]{text}{*name*}

creates a bookmark with the specified text and at the given level (default is 0). As name for the internal anchor name is used (in conjunction with level). Therefore the name must be unique (similar to \label).

    \currentpdfbookmark{*text*}{*name*}

creates a bookmark at the current level.

    \subpdfbookmark{*text*}{*name*}

creates a bookmark one step down in the bookmark hierarchy. Internally the current level is increased by one.

    \belowpdfbookmark{*text*}{*name*}

creates a bookmark below the current bookmark level. However after the command the current bookmark level has not changed.
Hint: Package bookmark replaces hyperref's bookmark organization by a new algorithm:

- Usually only one LaTeX run is needed.

- More control over the bookmark appearance (color, font).

- Different bookmark actions are supported (external file links, URLs, …).

Therefore I recommend using this package.

4.1.2   Replacement macros

hyperref takes the text for bookmarks from the arguments of commands like \section, which can contain things like math, colors, or font changes, none of which will display in bookmarks as is.

> \texorpdfstring{*T_EXstring*}{*PDFstring*}

For example,

```
\section{Pythagoras:
  \texorpdfstring{$ a^2 + b^2 = c^2 $}{%
   a\texttwosuperior\ + b\texttwosuperior\ =
   c\texttwosuperior
  }%
}
\section{\texorpdfstring{\textcolor{red}}{}{Red} Mars}
```

\pdfstringdef executes the hook before it expands the string. Therefore, you can use this hook to perform additional tasks or to disable additional commands.

```
\expandafter\def\expandafter\pdfstringdefPreHook
\expandafter{%
 \pdfstringdefPreHook
 \renewcommand{\mycommand}[1]{}%
}
```

However, for disabling commands, an easier way is via \pdfstringdefDisableCommands, which adds its argument to the definition of \pdfstringdefPreHook ('@' can here be used as letter in command names):

```
\pdfstringdefDisableCommands{%
 \let~\textasciitilde
 \def\url{\pdfstringdefWarn\url}%
 \let\textcolor\@gobble
}
```

4.2   Utility macros

> \hypercalcbp{*dimen specification*}

\hypercalcbp takes a TeX dimen specification and converts it to bp and returns the number without the unit. This is useful for options pdfview, pdfstartview and pdfremotestartview. Example:

```
    \hypersetup{
     pdfstartview={FitBH \hypercalcbp{\paperheight-\topmargin-1in
       -\headheight-\headsep}
    }
```

The origin of the PDF coordinate system is the lower left corner.

Note, for calculations you need either package |calc| or ε-TeX. Nowadays the latter should automatically be enabled for LaTeX formats. Users without ε-TeX, please, look in the source documentation hyperref.dtx for further limitations.

Also \hypercalcbp cannot be used in option specifications of \documentclass and \usepackage, because LaTeX expands the option lists of these commands. However package hyperref is not yet loaded and an undefined control sequence error would arise.

5   Acrobat-specific behavior

If you want to access the menu options of Acrobat Reader or Exchange, the following macro is provided in the appropriate drivers:

> `\Acrobatmenu{`*menuoption*`}{`*text*`}`

The *text* is used to create a button which activates the appropriate *menuoption*. The following table lists the option names you can use—comparison of this with the menus in Acrobat Reader or Exchange will show what they do. Obviously some are only appropriate to Exchange.

File	Open, Close, Scan, Save, SaveAs, Optimizer:SaveAsOpt, Print, PageSetup, Quit
File→Import	ImportImage, ImportNotes, AcroForm:ImportFDF
File→Export	ExportNotes, AcroForm:ExportFDF
File→DocumentInfo	GeneralInfo, OpenInfo, FontsInfo, SecurityInfo, Weblink:Base, AutoIndex:DocInfo
File→Preferences	GeneralPrefs, NotePrefs, FullScreenPrefs, Weblink:Prefs, AcroSearch:Preferences(Windows) or, AcroSearch:Prefs(Mac), Cpt:Capture
Edit	Undo, Cut, Copy, Paste, Clear, SelectAll, Ole:CopyFile, TouchUp:TextAttributes, TouchUp:FitTextToSelection, TouchUp:ShowLineMarkers, TouchUp:ShowCaptureSuspects, TouchUp:FindSuspect, Properties
Edit→Fields	AcroForm:Duplicate, AcroForm:TabOrder
Document	Cpt:CapturePages, AcroForm:Actions, CropPages, RotatePages, InsertPages, ExtractPages, ReplacePages, DeletePages, NewBookmark, SetBookmarkDest, CreateAllThumbs, DeleteAllThumbs
View	ActualSize, FitVisible, FitWidth, FitPage, ZoomTo, FullScreen, FirstPage, PrevPage, NextPage, LastPage, GoToPage, GoBack, GoForward, SinglePage, OneColumn, TwoColumns, ArticleThreads, PageOnly, ShowBookmarks, ShowThumbs
Tools	Hand, ZoomIn, ZoomOut, SelectText, SelectGraphics, Note, Link, Thread, AcroForm:Tool, Acro_Movie:MoviePlayer, TouchUp:TextTool, Find, FindAgain, FindNextNote, CreateNotesFile
Tools→Search	AcroSrch:Query, AcroSrch:Indexes, AcroSrch:Results, AcroSrch:Assist, AcroSrch:PrevDoc, AcroSrch:PrevHit, AcroSrch:NextHit, AcroSrch:NextDoc
Window	ShowHideToolBar, ShowHideMenuBar, ShowHideClipboard, Cascade, TileHorizontal, TileVertical, CloseAll
Help	HelpUserGuide, HelpTutorial, HelpExchange, HelpScan, HelpCapture, HelpPDFWriter, HelpDistiller, HelpSearch, HelpCatalog, HelpReader, Weblink:Home
Help(Windows)	About

6   PDF and HTML forms

You must put your fields inside a `Form` environment (only one per file).

There are six macros to prepare fields:

\TextField[*parameters*]{*label*}

\CheckBox[*parameters*]{*label*}

\ChoiceMenu[*parameters*]{*label*}{*choices*}

\PushButton[*parameters*]{*label*}

\Submit[*parameters*]{*label*}

\Reset[*parameters*]{*label*}

The way forms and their labels are laid out is determined by:

\LayoutTextField{*label*}{*field*}

\LayoutChoiceField{*label*}{*field*}

\LayoutCheckField{*label*}{*field*}

These macros default to #1 #2
What is actually shown in as the field is determined by:

\MakeRadioField{*width*}{*height*}

\MakeCheckField{*width*}{*height*}

\MakeTextField{*width*}{*height*}

\MakeChoiceField{*width*}{*height*}

\MakeButtonField{*text*}

These macros default to `\vbox to #2{\hbox to #1{\hfill}\vfill}`, except the last, which defaults to #1; it is used for buttons, and the special `\Submit` and `\Reset` macros.

You may also want to redefine the following macros:

```
\def\DefaultHeightofSubmit{12pt}
\def\DefaultWidthofSubmit{2cm}
\def\DefaultHeightofReset{12pt}
\def\DefaultWidthofReset{2cm}
\def\DefaultHeightofCheckBox{0.8\baselineskip}
\def\DefaultWidthofCheckBox{0.8\baselineskip}
\def\DefaultHeightofChoiceMenu{0.8\baselineskip}
\def\DefaultWidthofChoiceMenu{0.8\baselineskip}
\def\DefaultHeightofText{\baselineskip}
\def\DefaultHeightofTextMultiline{4\baselineskip}
\def\DefaultWidthofText{3cm}
```

6.1   Forms environment parameters

action	*URL*	The URL that will receive the form data if a Submit button is included in the form
encoding	*name*	The encoding for the string set to the URL; FDF-encoding is usual, and html is the only valid value
method	*name*	Used only when generating HTML; values can be post or get

6.2   Forms optional parameters

Note that all colors must be expressed as RGB triples, in the range 0..1 (i.e. color=0 0 0.5)

accesskey	key		(as per HTML)
align	number	*0*	alignment within text field; 0 is left-aligned, 1 is centered, 2 is right-aligned.
altname	name		alternative name, the name shown in the user interface
backgroundcolor			color of box
bordercolor			color of border
bordersep			box border gap
borderwidth		*1*	width of box border, the value is a dimension or a number with default unit bp
calculate			JavaScript code to calculate the value of the field
charsize	dimen		font size of field text
checkboxsymbol	char	*4 (✔)*	symbol used for check boxes (ZapfDingbats), the value is a character or \ding{*number*}, see package 'pifont' from bundle 'psnfss'
checked	boolean	*false*	whether option selected by default
color			color of text in box
combo	boolean	*false*	choice list is 'combo' style
default			default value
disabled	boolean	*false*	field disabled
format			JavaScript code to format the field
height	dimen		height of field box
hidden	boolean	*false*	field hidden
keystroke			JavaScript code to control the keystrokes on entry

`mappingname`	name		the mapping name to be used when exporting the field data
`maxlen`	number	*0*	number of characters allowed in text field
`menulength`	number	*4*	number of elements shown in list
`multiline`	boolean	*false*	whether text box is multiline
`name`	name		name of field (defaults to label)
`onblur`			JavaScript code
`onchange`			JavaScript code
`onclick`			JavaScript code
`ondblclick`			JavaScript code
`onfocus`			JavaScript code
`onkeydown`			JavaScript code
`onkeypress`			JavaScript code
`onkeyup`			JavaScript code
`onmousedown`			JavaScript code
`onmousemove`			JavaScript code
`onmouseout`			JavaScript code
`onmouseover`			JavaScript code
`onmouseup`			JavaScript code
`onselect`			JavaScript code
`password`	boolean	*false*	text field is 'password' style
`popdown`	boolean	*false*	choice list is 'popdown' style
`radio`	boolean	*false*	choice list is 'radio' style
`radiosymbol`	char	*H (★)*	symbol used for radio fields (ZapfDingbats), the value is a character or `\ding{`*number*`}`, see package 'pifont' from bundle 'psnfss'
`readonly`	boolean	*false*	field is readonly
`rotation`	number	*0*	rotation of the widget annotation (degree, counterclockwise, multiple of 90)
`tabkey`			(as per HTML)
`validate`			JavaScript code to validate the entry
`value`			initial value
`width`	dimen		width of field box

7 Defining a new driver

A hyperref driver has to provide definitions for eight macros:

1. `\hyper@anchor`
2. `\hyper@link`
3. `\hyper@linkfile`
4. `\hyper@linkurl`
5. `\hyper@anchorstart`
6. `\hyper@anchorend`
7. `\hyper@linkstart`
8. `\hyper@linkend`

The draft option defines the macros as follows

```
\let\hyper@@anchor\@gobble
\gdef\hyper@link##1##2##3{##3}%
\def\hyper@linkurl##1##2{##1}%
\def\hyper@linkfile##1##2##3{##1}%
```

```
\let\hyper@anchorstart\@gobble
\let\hyper@anchorend\@empty
\let\hyper@linkstart\@gobbletwo
\let\hyper@linkend\@empty
```

8 Special support for other packages

hyperref aims to cooperate with other packages, but there are several possible sources for conflict, such as

- Packages that manipulate the bibliographic mechanism. Peter William's harvard package is supported. However, the recommended package is Patrick Daly's natbib package that has specific hyperref hooks to allow reliable interaction. This package covers a very wide variety of layouts and citation styles, all of which work with hyperref.

- Packages that typeset the contents of the \label and \ref macros, such as showkeys. Since the hyperref package redefines these commands, you must set implicit=false for these packages to work.

- Packages that do anything serious with the index.

The hyperref package is distributed with variants on two useful packages designed to work especially well with it. These are xr and minitoc, which support crossdocument links using LaTeX's normal \label/\ref mechanisms and per-chapter tables of contents, respectively.

9 History and acknowledgments

The original authors of hyperbasics.tex and hypertex.sty, from which this package descends, are Tanmoy Bhattacharya (`tanmoy@qcd.lanl.gov`) and Thorsten Ohl (`thorsten.ohl@physik.th-darmstadt.de`). hyperref started as a simple port of their work to LaTeX 2_ε standards, but eventually I rewrote nearly everything, because I didn't understand a lot of the original, and was only interested in getting it to work with LaTeX. I would like to thank Arthur Smith, Tanmoy Bhattacharya, Mark Doyle, Paul Ginsparg, David Carlisle, T. V. Raman and Leslie Lamport for comments, requests, thoughts and code to get the package into its first useable state. Various other people are mentioned at the point in the source where I had to change the code in later versions because of problems they found.

Tanmoy found a great many of the bugs, and (even better) often provided fixes, which has made the package more robust. The days spent on RevTeX are entirely due to him! The investigations of Bill Moss (`bmoss@math.clemson.edu`) into the later versions including native PDF support uncovered a good many bugs, and his testing is appreciated. Hans Hagen (`pragma@pi.net`) provided a lot of insight into PDF.

Berthold Horn provided help, encouragement and sponsorship for the dvipsone and dviwindo drivers. Sergey Lesenko provided the changes needed for dvipdf, and Hàn Thê Thành supplied all the information needed for pdftex. Patrick Daly kindly updated his natbib package to allow easy integration with hyperref. Michael Mehlich's hyper package (developed in parallel with hyperref) showed me solutions for some problems. Hopefully the two packages will combine one day.

The forms creation section owes a great deal to: T. V. Raman, for encouragement, support and ideas; Thomas Merz, whose book *Web Publishing with Acrobat/PDF* provided crucial insights; D. P. Story, whose detailed article about pdfmarks and forms solved many practical problems; and Hans Hagen, who explained how to do it in pdftex.

Steve Peter recreated the manual source in July 2003 after it had been lost.

Especial extra thanks to David Carlisle for the `backref` module, the ps2pdf and dviwindo support, frequent general rewrites of my bad code, and for working on changes to the `xr` package to suit `hyperref`.

10   GNU Free Documentation License

Version 1.2, November 2002

Copyright © 2000,2001,2002 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

10.1   Applicability and definitions

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the

document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

10.2   Verbatim copying

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 10.3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

10.3   Copying in quantity

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

10.4   Modifications

You may copy and distribute a Modified Version of the Document under the conditions of sections 10.2 and 10.3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.

C. State on the Title page the name of the publisher of the Modified Version, as the publisher.

D. Preserve all the copyright notices of the Document.

E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

H. Include an unaltered copy of this License.

I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.

M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.

N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.

O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties–for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

10.5   Combining documents

You may combine the Document with other documents released under this License, under the terms defined in section 10.4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

10.6   Collections of documents

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

10.7   Aggregation with independent works

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 10.3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

10.8   Translation

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 10.4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 10.4) to Preserve its Title (section 10.1) will typically require changing the actual title.

10.9   Termination

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10.10   Future revisions of this license

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

Addendum: how to use this license for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

> Copyright © YEAR YOUR NAME. Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with...Texts." line with this:

with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.