Hypertext capabilities with pdfLaTeX

Federico Garcia
`federook (at) gmail dot com`

1 Introduction

With the standardization of electronic publishing and sharing of manuscripts, a wealth of new technical resources and possibilities is open to authors, resources that go beyond habitual on-paper typographic uses, to involve the new everyday tool of the reader: the mouse click.

The TeX world, as usual, has been quick to take up the new possibilities. In particular, pdfTeX by Hàn Thế Thành et al., along with its companion pdfLaTeX, and the hyperref LaTeX package by Sebastian Rahtz and Heiko Oberdiek — both available in standard distributions of TeX and LaTeX — are especially successful implementations of the possibilities of the PDF format — by far the most standardized, and by now ubiqitious, electronic format.

This article is intended as a quick guide to the most immediately usable functions of these tools. It is not intended to replace the available documentation, which is much more precise and complete. In fact, my intention is to steer away from completeness. For example, the file `options.pdf` in the hyperref distribution lists all the options to the package. This list, containing no explanations or running text, is enough to fill more than two pages — there's well over 60 options. The main fact about them, however, is that most of them are completely meaningless to the average user, who probably won't understand them anyway (at least in my case).

In other words, only a small portion of the universe of possibilities of on-screen documents is directly relevant for the average author. But the documentation of these features (the relevant ones) is all too often obscured by the rest of them. The whole business appears more overwhelming than it actually is. Hopefully, this article will help this perceived situation. It is an exposition of the (relevant) extended possibilities of pdfLaTeX in terms of the LaTeX we all know. In some cases this will actually imply a lie, a white one. A footnote will state as much in such cases, without going into further details.

2 Immediate special effects of hyperref

When hyperref is loaded (for useful options, and tips on loading the package, see Hypersetup and options) in an otherwise normal LaTeX document, a number of things happen without further intervention:

- The items in the table of contents, the list of figures, and the list of tables, will be links: when the reader clicks on them, the cursor will jump to the corresponding target.
- The superscript that calls for a footnote will be a link to the footnote itself.
- Bibliographical references through `\cite` will create links to the entries in the final bibliography list.
- All pairs of `\label-\ref` will also produce links (the result of a `\ref` leading by mouse click to the corresponding `\label`).

The last point merits a few notes. First of all, this is true of *all* elements capable of cross referencing in LaTeX: not only chapters and sections, but also to enumerate items, equations, and footnotes. For example, after

```
\footnote{This is the footnote.}\label{note}
\begin{enumerate}
\item \label{item1} This is the first item.
\item \label{item2} This, the second.
\end{enumerate}
```

a `\ref` to any of these keys will produce the corresponding number as a link.

The same is true for `\pageref`, which prints the number of the *page* where the referenced element appears (rather than the number of the element itself). With hyperref, this page number will be a link.

For chapters and sections, in addition, hyperref offers a third command in the family: `\nameref`. Instead of printing the chapter or section number, `\nameref` will typeset *the name* of the chapter/section. This is much better and elegant than using the number if the document is intended to be read on the screen.

3 Arbitrary cross references

`\label` and `\ref` function in connection with LaTeX counters. But on-screen reading is not limited to refer to things that have a number. Thus, hyperref offers commands for the creation of cross references that are independent from counters. These are `\hypertarget` and `\hyperlink`, exact analogues of `\label` and `\ref` respectively.

Both of these commands, however, have a second argument:

```
\hypertarget{⟨key⟩}{⟨text⟩}   (like \label)
\hyperlink{⟨key⟩}{⟨text⟩}       (like \ref)
```

For `\hypertarget`, ⟨*text*⟩ is the destination of the user's click; for `\hyperlink`, it is the text of the link itself. The ⟨*text*⟩, in both cases, is any LaTeX box.

As an example, the following code creates a picture (from the file `logo.png`) that is the destination of a link:

```
\hypertarget{ref1}{\includegraphics{logo.png}}
```

The link itself, with the text 'see the logo', would be created with

```
\hyperlink{ref1}{see the logo}
```

4 External links

4.1 Cross referencing between files

The links made by `\label`-`\ref` and `\hypertarget`-`\hyperlink` work within a single file. Cross referencing *between* files is possible thanks to a third pair of commands provided by hyperref, described next.

This time, however, the commands are less than analogous to the usual `\label`-`\ref`. They require not one, but two 'keys', called ⟨*key*⟩ and ⟨*category*⟩ in the syntax below). Why this is so has to do with PDF syntax, but the LaTeX user can think of ⟨*category*⟩ as a second key.

```
\hyperdef{⟨category⟩}{⟨key⟩}
\hyperref{⟨file⟩}{⟨category⟩}{⟨key⟩}{⟨text⟩}
```

For example, the following line sets up a destination in the present file (note the space at the end of the line):

```
\hyperdef{xmpl}{dest}␣%
```

An external file would link to this destination (in the present file) with

```
\hyperref{hypertext.pdf}{xmpl}{dest}
        {to the destination}
```

4.2 Links to the web

The other kind of common external link is to a webpage. The command `\url` takes one argument — the destination's URL address — and creates a link to it (a click on it opens the system's browser on the requested page). For example, `\url{www.tug.org}` leads to the TUG web site.

A related kind of link is the 'mailto' link: with a click on it, the system opens the local email program and creates a new message to the indicated email address. This is done through the command:

```
\href{mailto:⟨email address⟩}{⟨link text⟩}
```

5 Bookmarks

The hyperref package handles virtually everything related to bookmarks (sometimes called "outlines") that the average user comes across. If instructed (see Hypersetup and options), the package will automatically compile the bookmark panel from the items in the table of contents. This includes sections, chapters, etc., and also the usual LaTeX results of `\addtocontents` and related commands.

There are two things that are not so direct: first, how to get a bookmark that does *not* correspond to an item in the TOC. For this, hyperref offers `\pdfbookmark[`⟨*level*⟩`]{`⟨*text*⟩`}{`⟨*key*⟩`}`.

The ⟨*level*⟩ is 0 for chapters, 1 for sections, etc., and -1 for `\part`s. The ⟨*text*⟩ is the text of the bookmark itself. The ⟨*key*⟩ doesn't really matter to us (it's another of those PDF-format-related requirements), except for the requirement that it be unique.

The destination of a bookmark created by such a `\pdfbookmark` command is the exact place where the command is issued. The bookmark itself will be appended, in the bookmark panel, in the position where the command is issued *with respect to other chapters, sections, etc.*

For example, the present article has a "Dummy Bookmark", which is created right here with

```
\pdfbookmark[2]{"Dummy Bookmark"}{"bmkey"}
```

As a result, it appears (in the bookmark panel) right below the one for this section ("Bookmarks") and before the one for the next ("Text and TeX"). The fact that it appears as a subitem of "Bookmarks" is due to the `[2]` argument (the ⟨*level*⟩).

So, how to get bookmarks appended to *other* places in the panel? For example, how to create a bookmark whose destination comes after a section heading, but with the bookmark itself being placed before the one for that section in the bookmark panel?

This question is not simply a puzzle, but has a potentially useful application: the mapping of the list of figures or the list of tables (as well as the table of contents) in the bookmark panel. This is the second thing that is not so direct with hyperref (or pdfLaTeX, for that matter).

In fact, it is a relatively hard thing to achieve. I found a solution when writing pittetd, the electronic dissertations class at the University of Pittsburgh. I have a project of abstracting this part of the pittetd code and uploading it to CTAN as a small, independent package, but for the moment interested readers can consult the documentation of the solution in `pittetd.dtx`.

6 Text and TEXt

Bookmarks are the main environment of another issue PDF writers should be aware of: they are made only of plain text, and cannot support what LaTeX can put in — for example — section titles. For example, the next subsection:

6.1 α-expressions and H_2O

This heading looks right in the text, but the corresponding bookmark is wrong.

The alternative takes care of the bookmark:

6.2 Alpha-expressions and water

But the heading in the text is not satisfactory.

6.3 α-expressions and H_2O

The solution is to use another command from hyperref, this one named `\texorpdfstring`:

`\texorpdfstring{`⟨*TEX text*⟩`}{`⟨*plain text*⟩`}`

The result is a flexible expression that behaves like TEX text (TEXt) in a LaTeX context, and like plain text in PDF-related strings.

For example, the present subsection was created with:

```
\subsection{%
  \texorpdfstring{$\alpha$}{Alpha}-expressions
  and \texorpdfstring{H$_2$O}{water}}
```

This procedure is generally not needed with accents and commands with immediate expansion, including common logos and symbols. But hyperref cannot convert more advanced stuff — notably math mode — without help, and gives a warning. It is in these cases that `\texorpdfstring` is useful.

7 Post-it notes

On page 130 above there is an expandable 'post-it' note. In my opinion, this easily overlooked resource of PDF offers an excellent alternative to the footnote when a document is intended to be read on the screen. hyperref does not include support for these notes, but their creation (through pdfTEX primitives) is not hard:

```
  \pdfannot width w height h depth d
  { /Subtype /Text /Contents (text) }
```

The three dimensions `w`, `h`, and `d` are all LaTeX dimensions. But the one that is important is `h` (the height), because it determines whe[illegible]e note appears in relation to the text baseline. [illegible]is a good idea use a `\qquad` after the command.

One thing to keep in mind with post-it notes is that their exact behavior (color, size, when it opens, how it closes, etc.) is not very standardized, and tends to change from version to version of Acrobat Reader. Nevertheless, the note can be given a different color if, between the braces of `\pdfannot`, `/C [r g b]` is appended; a title for the note is determined with `/T (title)`; an[illegible] note can be open by default with `/Open true`.

If post-it notes are used at all, the document has to be typeset by pdfTEX (rather than converted from DVI), since otherwise the primitive `\pdfannot` is not available. See the next section.

8 Hypersetup and options

The hyperref package has so many options that it provides a separate command `\hypersetup` for configuration. Thus, options to the package can be specified either in the usual way

`\usepackage[`⟨*options*⟩`]{hyperref}`

or in a separate line:

`\hypersetup{`⟨*option, option, ...*⟩`}`

Some of the many options of hyperref are of particular interest.

```
colorlinks
linkcolor=red
urlcolor=blue
citecolor=green
```

```
bookmarks
bookmarksopen
bookmarksnumbered=false
```

```
pdftitle="Hypertext capabilities with pdfTeX"
pdfauthor="Federico Garcia"
pdfkeywords="Bookmarks, links, PDF, LaTeX"
```

The last three are useful to set up the fields of the 'Document Properties' information. (Again, the behavior of this is not reliably standard from version to version of Acrobat Reader.)

Another thing to keep in mind is that hyperref has to know the way the PDF file is produced: whether pdfTEX is run on the document, or a separate program will convert the DVI — in which case hyperref needs to load the corresponding driver. This information is given to the package as an option ('`pdftex`' for pdfTEX, '`dvipdfm`' if this program is used, etc.). Since this information is also used by other packages — notably graphicx — it is customary to indicate this option as a general option *to the document class*, which will then pass it to any package that needs it.

9 A tip

hyperref changes the internal mechanism of cross references. The change is truly magical, in the sense that the user, most of the time, has and needs to have no idea that a change occurred.

However, there is one case in which the change becomes relevant: when a document is 'converted' from plain to hyperlinked, or the reverse; in other words, when the `\usepackage{hyperref}` line is added or removed. The presence of auxiliary files created by one mechanism and used by the other *will* create quite unintelligible error messages.

Therefore, a tip: make sure to delete all auxiliary files (`.aux`, `.toc`, `.lot`, `.lof`, ...) when you will run a document for the first time with hyperref (or, later, for the first time without it).

10 Material not covered

Two main things are not covered in this document: 'hyper-bibliography', in which the entries in the bibliography list can link back to the citations in the text; and 'hyper-index', in which the page numbers in the index are links to the corresponding pages.

These topics are not so easy to deal with in a general manner. In the main, support for these tends to be fragile, because of the myriad of styles for both bibliography and index. As a result, dealing with them essentially requires proficiency with BibTeX and *MakeIndex*. Given this proficiency, the file `hyperref.dtx` is a source of information on how to get things done.

In the case of bibliographies, a more immediate guide is the file `backref.pdf`, actually the documentation to the sub-package of the same name (by David Carlisle and Sebastian Rahtz). The file is part of the `doc` directory of hyperref.

11 Documentation of hyperref

This is a description of the PDF files in the `doc` directory of hyperref:

manual by Sebastian Rahtz is a reference guide to the package. Not very useful for beginners, it contains often crucial information on details, important when you are doing something extraordinary.

paper by Heiko Oberdiek is a precise and complete explanation of many of the topics treated here, from the point of view of 'how does it work?' (rather than 'how do I use it?').

slides is the set of slides used by Heiko in his presentation of '`paper`'. It is an illustration of the possibilities of using hyperref, pdfTeX, and the package thumbpdf.