

texlinks.sty

TeX-Related Links for `hyperref`, `blog.sty` (and maybe more)*

Uwe Lück[†]

January 28, 2011

Abstract

`texlinks.sty` provides a couple of shorthands for making hyperlinks with `hyperref`'s `\href` command, linking to URLs that one often refers to in discussing TeX-related material. Especially, TUG material (including texhax postings and TUGboat articles) and CTAN pages (package descriptions, directories, Catalogue) are supported, also the UK FAQ and Wikipedia (where much TeX-related software is described in a visually appealing manner). However, up to now I have used them for *HTML* overviews generated with `blog.sty`. They may as well be useful with better known (and better developed) TeX \rightarrow HTML software such as `tex4ht` or `LaTeX2HTML` (I don't know).

Contents

1	Usage	2
2	Package File Header (Legalize)	2
3	Outline	3
4	Package Options	3
5	Providing <code>\httpref</code>	3
6	Variants of <code>\httpref</code>	3
7	Wikipedia	5

*This document describes version [v0.2](#) of `texlinks.sty` as of 2011/01/27.

[†]<http://contact-ednotes.sty.de.vu>

1	USAGE	2
8	T_EX-related	6
8.1	CTAN	6
8.2	TUG	7
8.2.1	texhax	7
8.2.2	Other	8
8.3	UK FAQ	8
9	Leaving	8
10	VERSION HISTORY	8

1 Usage

The file `texlinks.sty` is provided ready, installation only requires putting it somewhere where T_EX finds it (which may need updating the filename data base).¹

Below the `\documentclass` line(s) and above `\begin{document}`, you load `texlinks.sty` (as usually) by

```
\usepackage{texlinks}
```

Package options and user commands are described near their definitions below in the implementation section.

2 Package File Header (Legalize)

```

1 \NeedsTeXFormat{LaTeX2e}[1994/12/01] %% \newcommand* etc.
2 \ProvidesPackage{texlinks}[2011/01/27 v0.2 TeX-related links (UL)]
3 %% copyright (C) 2011 Uwe Lueck,
4 %% http://www.contact-ednotes.sty.de.vu
5 %% -- author-maintained in the sense of LPPL below.
6 %%
7 %% This file can be redistributed and/or modified under
8 %% the terms of the LaTeX Project Public License; either
9 %% version 1.3c of the License, or any later version.
10 %% The latest version of this license is in
11 %% http://www.latex-project.org/lppl.txt
12 %% We did our best to help you, but there is NO WARRANTY.
13 %%
14 %% Please report bugs, problems, and suggestions via
15 %%
16 %% http://www.contact-ednotes.sty.de.vu
17 %%

```

¹<http://www.tex.ac.uk/cgi-bin/texfaq2html?label=inst-wlcf>

3 Outline

All the link macros of `texlinks` are based on a macro `\httpref`. For use of `texlinks` with `blog.sty`, the latter provides a definition of `\httpref` suitable for HTML, where a choice of opening a new tab or window—or not—is relevant.

For use with `hyperref` (or `...?`), `texlinks` may provide a definition of `\httpref` based on `\href`. The decision to do so or not may happen at `\begin{document}`. `blog.sty` generates HTML without using the `{document}` environment, so we might assume that when `\begin{document}` is found, we are running `hyperref`, or just *something* that provides a useful `\href`. We might then execute a definition of `\httpref` in terms of `\href`. Well, not sure ...

Moreover, a PDF file with links may be *printed*, and clicking the links on the paper may fail. URLs in main text, on the other hand, sometimes are troublesome. I consider it a good idea to present links with their URL as the displayed text in *footnotes* (or *endnotes*). It may even be useful with HTML to present the URLs displayed in some 2011/01/27: “appendix.”—This idea has been resumed in v0.2 only, `\urlfoot`.

4 Package Options

Somebody may want to suppress a definition of `\httpref` at `\begin{document}` ... [2011/01/24, [TODO](#)]

5 Providing `\httpref`

`\httpref{<host-path/#frag>}{<text>}`

should display `<text>` as a link to `http://<host-path/#frag>`. In case `\begin{document}` is found with a definition of `\href` present, we provide a definition of `\httpref` in terms of `\href` there:

```
18 \AtBeginDocument{%
19   \@ifundefined{href}{%
20     % \PackageError ... TODO!? 2011/01/24
21     }\newcommand*{\httpref}[1]{\href{http://#1}}}
```

6 Variants of `\httpref`

`\NormalHTTPref` may be used as an alias for `\httpref` in situations where the latter has been redefined:

```
22 \AtBeginDocument{%% TODO: options, guarded \let (mine, HO)
23   \@ifdefinable\NormalHTTPref{\let\NormalHTTPref\httpref}}
```

`\ithttpref{<url>}{<text>}` displays `<text>` in italics:

```
24 \newcommand*{\ithttpref}[2]{\NormalHTTPref{#1}{\textit{#2}}}
```

With `\urlhttpref{<url>}`, that URL is displayed:

```
25 \newcommand*{\urlhttpref}[1]{%
26   \NormalHTTPref{#1}{\urlfmt{\httpprefix#1}}}
```

`\httpprefix` is an idea that was missing in `blog.sty` up to v0.3. It may be used to determine generally whether a display of an URL should include `http://`. I choose as default what was default in `blog.sty` (i.e., “don’t include”):

```
27 \ifdefinable\httpprefix{\let\httpprefix\empty}    %% TODO cf. above
\let\httpprefix\relax would be bad for blog.sty (would display \relax),
while it would be somewhat more efficient.
```

Now you may customize `\httpprefix` by

```
\renewcommand{\httpprefix}{http://}
```

—or by `\let\httpprefix\theHTTPprefix`:

```
28 \newcommand*{\theHTTPprefix}{http://}
```

`\urlfmt{<url>}` is chosen as `\texttt` here and may be customized, e.g., with the `breakurl` package.

```
29 \ifdefinable\urlfmt{\let\urlfmt\texttt}
```

In `blog.sty` (as of 2010/05/26), there was a command `\urlref` instead of `\urlhttpref`. It did not provide `\urlfmt`.

`\foothttpurlref{<url>}` just is like `\footnote{\urlhttpref{<url>}}`:

```
30 \newcommand*{\foothttpurlref}[1]{\footnote{\urlhttpref{#1}}}
```

`\urlfoot{<short>}{<id>}` redefines `\httpref` so that you can use all the shorthand macros based on `\httpref` to get the according URL display (as provided by `\urlhttpref`) in a footnote without the need to include the entire URL in your source code. `\urlfoot` is available with `<short>` and `<id>` when a shorthand `\<short>{<id>}{<text>}` has been defined where `\<short>` is the macro name and `<id>` is the target identifier (usually part of the URL generated from `<id>`) according to the syntax declaration of `\<short>`.

```
31 \newcommand*{\urlfoot}[2]{%
32   \let\httpref\foothttpurlref
33   \let\httpprefix\theHTTPprefix    %% TODO customizable!?
34   \csname #1\endcsname{#2}{}}
```

Example:

```
\CtanPkgRef{morehype}{MoreHype} and \ctanpkgref{morehype}
```

are provided below for linking to `http://ctan.org/pkg/morehype`.

- Try `CtanPkgRef` *here*: `MoreHype`,
for the *footnote* try `\urlfoot{CtanPkgRef}{morehype}`;²
- try `ctanpkgref` *here*: `morehype`,
for the *footnote* try `\urlfoot{ctanpkgref}{morehype}`.³`morehype`

²<http://ctan.org/pkg/morehype>

³<http://ctan.org/pkg/morehype>

The lonely ‘morehype’ demonstrates that it doesn’t work with `\ctanpkgref` because `\ctanpkgref` doesn’t have separate arguments for $\langle id \rangle$ and $\langle text \rangle$, it actually doubles $\langle id \rangle$. A local `\let\ctanpkgref\ctanPkgRef` could help, but right now I prefer waiting for a better idea. [TODO]

7 Wikipedia

`\Wikideref{ $\langle lemma \rangle$ }` refers to article $\langle lemma \rangle$ in the German Wikipedia. (The next considerations apply to English etc. as well!) With umlauts etc., `\wikideref{ $\langle wiki-id \rangle$ }{ $\langle lemma \rangle$ }` may be needed. For this form, sometimes using ‘_’ seemed to be needed. In `blog.sty`, there was `\catcode‘_’=12`. This was no problem there, as the underscore was not needed for math there. (For some time I thought of making it `\active` for a shorthand for `<sub>` elements ...) But I could not find out whether it was needed or avoidable. I leave this matter open here (for `hyperref`) ... [2011/01/24, TODO, maybe solved there]

```
35 \newcommand*{\wikideref}[1]{\httpref{de.wikipedia.org/wiki/#1}}
36 \newcommand*{\Wikideref}[1]{\wikideref{#1}{#1}}
```

Anyway, `\underscorechar` seems to be useful in our macro definitions. The name is inspired by L^AT_EX’s `\@backslashchar` and `\@percentchar`:

```
37 \newcommand \underscorechar {}
38 {\@makeother\_ \gdef\underscorechar{\_}}
```

Note: It would be natural to replace `\Wikideref` by equipping `\wikideref` with an optional argument—with L^AT_EX. With `blog.sty` however, where this code is now taken from, optional arguments didn’t work. Later we might indeed in a similar manner to dealing with `hyperref` at `\begin{document}` introduce optional arguments for the case of typesetting.—Analogue applies in the sequel again. [2011/01/24, TODO]

`\itwikideref` is an italic variant of `\wikideref`:

```
39 \newcommand*{\itwikideref}[2]{\wikideref{#1}{\textit{#2}}}
```

`\Wikienref{ $\langle wiki-id \rangle$ }{ $\langle lemma \rangle$ }` refers to the English Wikipedia:

```
40 \newcommand*{\wikienref}[1]{\httpref{en.wikipedia.org/wiki/#1}}
```

By analogy to `\Wikideref`, `\Wikienref{ $\langle lemma \rangle$ }` saves you from repeating $\langle lemma \rangle$:

```
41 \newcommand*{\Wikienref}[1]{\wikienref{#1}{#1}}
```

Quite often, programs share their names with movies, biological species, etc., then disambiguation is required. Usually, the movie is irrelevant and we don’t want to display the disambiguation. The following macros save you from typing the underscore (didn’t spaces suffice sometimes?) and the round parentheses. `\wikiendisambref{ $\langle term \rangle$ }{ $\langle suffix \rangle$ }{ $\langle text \rangle$ }` links to

`http://en.wikipedia.org/wiki/` $\langle term \rangle$ `_` $\langle suffix \rangle$

while displaying $\langle text \rangle$ as link text only:

```

42 \newcommand*\wikiendisambref}[2]{%
43   \wikienref{#1\underscorechar (#2)}}

```

`\Wikiendisambref{⟨term⟩}{⟨suffix⟩}` does the same when `⟨text⟩` is the same as `⟨term⟩`:

```

44 \newcommand*\Wikiendisambref}[2]{%
45   \wikienref{#1\underscorechar (#2)}{#1}}

```

Italic variant `\itwikienref` of `\wikienref` (blog.sty had `\emwikienref` instead):

```

46 \newcommand*\itwikienref}[2]{\wikienref{#1}{\textit{#2}}}

```

`\urluml{⟨ascii-char⟩}` as of 2010/05/25 (not sure if it ever worked or was actually needed):

```

47 \newcommand*\urluml}[1]{\csname urluml:#1\endcsname}
48 \@namedef{urluml:a}{\#C3\#A4}
49 \@namedef{urluml:o}{\#C3\#B6}
50 \@namedef{urluml:u}{\#C3\#BC}
51 \@namedef{urluml:s}{\#C3\#9F}          %% 2010/08/09

```

For **anchors**, ‘#’ can be used with `blog.sty`—and even with `hyperref`.

Example: `\wikienref{TeX#History}{TeX}` for T_EX.

8 T_EX-related

8.1 CTAN

`\bytopicref{⟨anchor⟩}{⟨text⟩}` makes `⟨text⟩` a link to `⟨anchor⟩` of Jürgen Fenn’s Topical Index of the T_EX Catalogue. You find the `⟨anchor⟩` by clicking at the respective TOC entry on top of the page and then read the URL from the browser’s navigation display.

```

52 \newcommand*\bytopicref}[1]{%
53   \httpref{mirror.ctan.org/help/Catalogue/bytopic.html\##1}}

```

(Example: `\bytopicref{html}{HTML}` for HTML.)

`\tugctanref{⟨path⟩}{⟨text⟩}` makes `⟨text⟩` a link to a T_EX Archive directory or file `⟨path⟩`:

```

54 \newcommand*\tugctanref}[1]{%
55   \httpref{tug.ctan.org/tex-archive/#1}}

```

`\ctanpkgref{⟨pkg-name⟩}` makes `⟨text⟩` a link to the CTAN package info page for the package `⟨pkg-name⟩`. `\CtanPkgRef{⟨name⟩}{⟨Name⟩}` is a variant for the cases where authors have a special idea `⟨Name⟩` using some capital letters when they describe their packages (ASCII versions of “Logos” such as BibT_EX) while the identifier `⟨name⟩` doesn’t allow capital letters. Also, `⟨Name⟩` may be a package from a *bundle* `⟨name⟩` where `⟨name⟩` has a description page while `⟨Name⟩` doesn’t have its *own* description page (such as `fifinddo`).

```

56 \newcommand*{\CtanPkgRef}[2]{%
57   \httpref{ctan.org/pkg/#1}{\pkgnamefmt{#2}}}
58 \newcommand*{\ctanpkgref}[1]{\CtanPkgRef{#1}{#1}}

```

Instead of `\pkgnamefmt{<pack-name>}`, `blog.sty` had `\prg{<pack-name>}`, without a proper implementation. For typesetting, choosing `\pkgnamefmt` as `\textsf` seems to conform to common practice today. The following code may later be suppressed at some package options, as with the choice for `\httpref`:

```

59 \@ifdefinable\pkgnamefmt {\let\pkgnamefmt\@firstofone}
60 \AtBeginDocument {\let\pkgnamefmt\textsf}

```

8.2 TUG

`\tugref{<path>}{<text>}` makes `<text>` a link to `<path>` on domain `tug.org`:

```

61 \newcommand*{\tugref}[1]{\httpref{tug.org/#1}}

```

8.2.1 texhax

`\texhaxref{<id>}{<text>}` makes `<text>` a link to the TUG web page displaying a texhax posting. You find `<id>` by searching `tug.org/pipermail/texhax/` and then reading the URL. `\THref{<id>}` saves you from choosing `<text>` and uses `texhax` instead.

```

62 \newcommand*{\texhaxref}[1]{\tugref{pipermail/texhax/#1}}
63 \newcommand*{\THref}[1]{\texhaxref{#1}{texhax}}

```

(It was `\prg{texhax}` in `blog.sty`, to have something logo-like, without a good idea how to implement it.)

`\texhaxpref{<id-code>}{<text>}` is a variant of `\texhaxref` where in place of `<id>` you only type the third and fourth digit of the year, then a `-`, then the (arabic) number of the month, then another `-`, and then actual internal identifier (a number of six digits preceding `.html` of the URL). I made this macro because I prefer typing to copying from the URL.

```

64 \newcommand*{\texhaxpref}[1]{%      %% 2010/09/07
65   \texhaxref{20\@texhax@parse#1/.html}}
66 \def\@texhax@parse#1-#2-#3/{%
67   #1-%
68   \ifcase #2\or
69     January\or February\or March\or   April\or
70     May\or June\or July\or August\or
71     September\or October\or November\or December% 2010/12/23
72   \fi
73   /#3}

```

TODO: `\texhaxPref#1` searches list of offsets to determine year/month from `id`

