

# texlinks.sty

---

## T<sub>E</sub>X-Related Links for `hyperref`, `blog.sty` (and maybe more)\*

Uwe Lück<sup>†</sup>

November 29, 2012

### Abstract

`texlinks.sty` provides a couple of shorthands for making hyperlinks with `hyperref`'s<sup>1</sup> `\href` command, linking to URLs that one often refers to in discussing T<sub>E</sub>X-related material. Especially, TUG material (including texhax postings and TUGboat articles) and CTAN pages (package descriptions, directories, Catalogue) are supported, also the UK FAQ, the L<sup>A</sup>T<sub>E</sub>X Wikibook, and Wikipedia (where much T<sub>E</sub>X-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) T<sub>E</sub>X → HTML software such as `tex4ht`<sup>2</sup> or `LaTeX2HTML`<sup>3</sup> (I don't know).

## Contents

1	Usage	2
2	Package File Header (Legalese)	3
3	Outline	3
4	Package Options	3
5	Fonts for URLs and File/Package Names	4

---

\*This document describes version **v0.7** of `texlinks.sty` as of 2012/11/28.

<sup>†</sup><http://contact-ednotes.sty.de.vu>

<sup>1</sup><http://ctan.org/pkg/hyperref>

<sup>2</sup><http://ctan.org/pkg/tex4ht>

<sup>3</sup><http://ctan.org/pkg/latex2html>

<b>1</b>	<b>USAGE</b>	<b>2</b>
<b>6</b>	<b>Providing \httpref and \httpsref</b>	<b>4</b>
<b>7</b>	<b>Variants of \httpref and \httpsref</b>	<b>5</b>
7.1	URLs as Links . . . . .	5
7.2	Linking URLs in Footnotes . . . . .	6
7.3	URL Bases . . . . .	6
<b>8</b>	<b>Wikipedia</b>	<b>7</b>
8.1	Backbones . . . . .	7
8.2	Piped Links . . . . .	8
8.3	English and German . . . . .	9
8.4	“Implicit” Choice of Language . . . . .	9
8.5	Blanks and Umlauts in URLs and Anchors . . . . .	10
<b>9</b>	<b>T<sub>E</sub>X-related</b>	<b>11</b>
9.1	CTAN . . . . .	11
9.1.1	Directories and Files in a T <sub>E</sub> X Archive . . . . .	11
9.1.2	Jürgen Fenn’s Topical T <sub>E</sub> X Catalogue . . . . .	12
9.1.3	Jim Hefferon’s Package Descriptions . . . . .	12
9.2	Mailing Lists . . . . .	13
9.3	CTAN Announcements . . . . .	14
9.4	TUG . . . . .	14
9.4.1	texhax . . . . .	15
9.4.2	Other . . . . .	15
9.5	UK FAQ . . . . .	16
9.6	Wikibooks . . . . .	16
<b>10</b>	<b>Leaving and Version HISTORY</b>	<b>16</b>

# 1 Usage

The file `texlinks.sty` is provided ready, installation only requires putting it somewhere where T<sub>E</sub>X finds it (which may need updating the filename data base).<sup>4</sup>

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.

---

<sup>4</sup><http://www.tex.ac.uk/cgi-bin/texfaq2html?label=inst-wlcf>

## 2 Package File Header (Legalese)

```

1 \NeedsTeXFormat{LaTeX2e}[1994/12/01] %% \newcommand* etc.
2 \ProvidesPackage{texlinks}[2012/11/28 v0.7 TeX-related links (UL)]
3 %% copyright (C) 2011 2012 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 %%

```

## 3 Outline

The link macros of `texlinks` are based on macros `\httpref` and `\httpsref`. For use of `texlinks` with `blog.sty`, the latter provides definitions of `\httpref` and `\httpsref` 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 definitions of `\httpref` and `\httpsref` 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 “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](#)]

v0.3: Package option `[blog]` suppresses *any* `\AtBeginDocument` actions—fine for use with `blog.sty`.

```

18 \DeclareOption{blog}{\let\AtBeginDocument\@gobble}

```

This option may be improved, and another option may be useful for different purposes than running `blog.sty`.

19 `\ProcessOptions`

## 5 Fonts for URLs and File/Package Names

This section “provides” markup for displaying URLs (`\urlfmt`), file names (`\filenamefmt`)—thinking of single files that may be found in the internet or on your computer—, and “packages” (`\pkgnamefmt`). For the latter two, in certain files I use shorthands `\file` and `\pkg`, resp., ... (Not sure about `\providecommand` `TODO` ...)

It is usual to use `\texttt` for formatting T<sub>E</sub>X code (“verbatim”, `\verb` etc.). It may also be common to use `\texttt` for file names, perhaps even for URLs. Therefore we provide `\urlfmt{<url>}` as follows:

20 `\providecommand*{\urlfmt}{\texttt}`

The file name format `\filenamefmt{<file-name>}` may differ from the format for URLs—if somebody wants/adjusts it, *here* it is the *same*:

21 `\providecommand*{\filenamefmt}{\texttt}`

... I favour `\code` over `\texttt` as “logical markup,” inspired by the `<code>` element in HTML, but it is too difficult to provide this right now here ...

`\pkgnamefmt{<package-name>}` displays the name of a “package”. Using `\textsf` for `\pkgnamefmt` seems to conform to common practice today—implemented here. The following code may later be suppressed at some package options, as with the choice for `\httpref`:

22 `% \ifdefinable\pkgnamefmt {\let\pkgnamefmt\@firstofone}`  
 23 `% \AtBeginDocument {\let\pkgnamefmt\textsf}`

← This was here until v0.7, makes a difference for PDF vs. `blog` HTML. Now we choose the same as with `\urlfmt`:

24 `\providecommand*{\pkgnamefmt}{\textsf}`

Indeed, the same day we are providing `\textsf` in `blog.sty`. However, the rationale of the earlier solution was that web pages use sans-serif as the *normal* font ...

## 6 Providing `\httpref` and `\httpsref`

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

should display `<text>` as a link to `http://<host-path[#frag]>`;

`\httpsref{<host-path[#frag]>}{<text>}`

is the obvious analogue for `https:` URLs. In case `\begin{document}` is found with a definition of `\href` present, we provide definitions of `\httpref` and `\httpsref` in terms of `\href` there:

```

25 \AtBeginDocument{%
26   \@ifundefined{href}{%
27     % \PackageError ... TODO!? 2011/01/24
28     }{\newcommand*\httpref}[1]{\href {http://#1}}%
29     \newcommand*\httpsref}[1]{\href {https://#1}}}
```

## 7 Variants of \httpref and \httpsref

`\NormalHTTPref` may be used as an alias for `\httpref` in situations where the latter has been redefined (as in Section 7.2):

```

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

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

```

32 % \newcommand*\ithttpref[2]{\NormalHTTPref{#1}{\textit{#2}}}
```

However, I seem never to have used it. And I would now prefer `\metahttpref`  
 TODO ...

### 7.1 URLs as Links

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

```

33 \newcommand*\urlhttpref[1]{%
34   \NormalHTTPref{#1}{\urlfmt{\httpprefix#1}}}
```

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

`\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”):

```

35 \@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`:

```

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

With `\urlhttpsref{<url>}`, we *force* displaying ‘`https://`’:

```

37 \newcommand*\urlhttpsref[1]{\httpsref{#1}{\urlfmt{https://#1}}}
```

## 7.2 Linking URLs in Footnotes

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

```
38 \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>`.

```
39 \newcommand*{\urlfoot}[2]{%
40   \let\httpref\foothttpurlref
41   \let\httpprefix\theHTTPprefix   %% TODO customizable!?
42   \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}`;<sup>5</sup>
- try `ctanpkgref` *here*: `morehype`,  
for the *footnote* try `\urlfoot{ctanpkgref}{morehype}`.<sup>6</sup>`morehype`

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

v0.3: Now that using `\urlfoot` and `ctanpkgref` together is so clumsy, while I use it quite often, we get `\urlpkgfoot{<package-id>}`, abbreviating `\urlfoot{CtanPkgRef}{<package-id>}`:

```
43 \newcommand* {\urlpkgfoot} {\urlfoot{CtanPkgRef}}
```

## 7.3 URL Bases

We typically refer to many web pages under a certain domain, or in certain subdirectories there. Before v0.6, I made many definitions like

```
\newcommand*{\myref}[1]{\httpref{<my-base>/#1}}
```

---

<sup>5</sup><http://ctan.org/pkg/morehype>

<sup>6</sup><http://ctan.org/pkg/morehype>

for this purpose. Storing the definition of such a `\myref` uses 6 tokens (including parameter text—[TODO](#)) in addition to those from `\my-base`. With

```
\newcommand*\myref{\httpbaseref{\my-base}}
```

we need 3 tokens instead, using `\httpbaseref{\base}{\rest}` defined as follows:

```
44 \newcommand*\httpbaseref[2]{\httpref{#1/#2}}
```

We change many definitions in ensuing sections accordingly.

The situation is similar with (many) anchors of a (large) web page. With v0.6, we introduce `\httppancref{\page-url}{\anchor}`—perhaps, with `\mirrorctanref` (Sec. 9.1.2) etc.? [TODO](#)

```
45 % \newcommand*\httppancref[2]{\httpref{#1\##2}}
```

## 8 Wikipedia

### 8.1 Backbones

As of v0.6, we have a backbone macro

```
\wikilangref{\language-code}{\lemma}{\text}
```

for links to Wikipedia. (It was `\wikiref` before, starting with v0.4—sorry!) `\language-code` consists of two characters like ‘de’ for German Wikipedia articles or ‘en’ for English ones. `\lemma` is the identifier of the article, and `\text` is displayed as the link:

```
46 % \newcommand*\wikilangref[2]{\httpref{#1.wikipedia.org/wiki/#2}}
```

← 2012/03/09 etc. with Sec. 7.3 →

```
47 \newcommand*\wikilangref[1]{\httpbaseref{#1.wikipedia.org/wiki}}
```

There is `\Wikilangref{\language-code}{\lemma}` for the case that `\lemma` and `\text` are the same. With v0.7 however, this command becomes more powerful, see Section 8.2.

```
48 % \newcommand*\Wikilangref[2]{\wikilangref{#1}{#2}{#2}}
```

`\wikilangref{\lang}{\lemma}{\text}` would be nicer; however, the present code is to work with `blog.sty` which does not support optional arguments.

Quite often, programs share their names with movies, biological species, etc., then lemma disambiguation is required. Usually, we don’t want to display the disambiguation.

```
\Wikilangdisambref{\language-code}{\term}{\suffix}
```

will link to

```
http://\language-code.wikipedia.org/wiki/\term_{\suffix}
```

```
49 \newcommand*\Wikilangdisambref}[3]{\wikilangref{#1}{#2 (#3)}{#2}}
```

There was something like a more general variant `\wikidisambref`, now I doubt its usefulness and `omit` it in order to see where it occurs (2011/05/13).

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

**Example:** `\wikilangref{en}{TeX#History}{\TeX}` for  $\TeX$ .

## 8.2 Piped Links

v0.7 emulates Wikipedia’s piped links as with Wikipedia source code

```
[[Pipeline|Pipe]]
```

to get a link to article ‘Pipeline’ with displayed text ‘Pipe’. The same syntax (double brackets) is actually supported by `blog.sty` with `blogexec.sty`, while otherwise only

```
\Wikilangref{<language-code>}{<lemma>|<text>}
```

works—with settings more below something like `\Wikiref{<lemma>|<text>}`—which admittedly is not much better than the equivalent

```
\wikiref{<lemma>}{<text>}
```

Even Wikipedia’s feature that empty `<text>` removes the disambiguation term as with `[[PipeL(computing)|]]` resulting in ‘Pipe’ is supported.

```
50 \newcommand*\Wikilangref}[2]{%
51   \@wikilpref{#1}#2\BiteSep|\@nnil\BiteSep\@nil{#2}}
```

I have introduced `\BiteSep` and this kind of parsing in the `bitelist`<sup>7</sup> package.

```
52 \def\@wikilpref#1#2|#3\BiteSep#4\@nil#5{%
53   \ifx\@nnil#3\@empty
54     \wikilangref{#1}{#5}{#5}%
55   \else
56     \wikilangref{#1}{#2}{%
57       \ifx\@three#3\@three
58         \wiki@noparen#2\@nil%
59       \else
60         #3%
61       \fi}%
62   \fi}
63 \def\wiki@noparen#1 (#2\@nil{#1}
```

I have thought about improving `bitelist.sty`, resulting in the following code. In the present application, I do not consider it superior. It uses the same number of tokens but new one has additional expansion step. The situation is different to the general case because doing everything before `\fi` is okay here.

<sup>7</sup><http://ctan.org/pkg/bitelist>



```

64 % \newcommand*{\Wikilangref}[2]{%
65 %     \@wikilPref{#1}#2\BiteSep\@secondoftwo
66 %     |\BiteSep\@firstoftwo\@nil{#2}}
67 % \def\@wikilPref#1#2|#3\BiteSep#4#5\@nil#6{%
68 %     #4{\wikilangref{#1}{#6}{#6}}%
69 %     {\wikilangref{#1}{#2}{%
70 %         \ifx\@three#3\@three
71 %         \wiki@noparen#2\@nil%
72 %         \else
73 %         #3%
74 %         \fi}}}

```

### 8.3 English and German

The next macros just save you from typing braces around the language codes for English and German: `\wikienref{<lemma>}{<text>}` refers to the English Wikipedia, `\wikideref{<lemma>}{<text>}` refers to the German one.

```

75 \newcommand*{\wikideref}{\wikilangref{de}}
76 \newcommand*{\wikienref}{\wikilangref{en}}

```

`\Wikideref{<lemma>}` refers to article `<lemma>` in the German Wikipedia and displays `<lemma>` as `<text>`:

```

77 \newcommand*{\Wikideref}{\Wikilangref{de}}

```

`\Wikienref{<lemma>}` is `\Wikideref`’s analogue for English:

```

78 \newcommand*{\Wikienref}{\Wikilangref{en}}

```

`\Wikidedisambref{<lemma>}{<suffix>}` chooses a disambiguation according to `<suffix>` for the German Wikipedia, `\Wikiendisambref{<lemma>}{<suffix>}` for the English one:

```

79 \newcommand*{\Wikidedisambref}{\Wikilangdisambref{de}}
80 \newcommand*{\Wikiendisambref}{\Wikilangdisambref{en}}

```

### 8.4 “Implicit” Choice of Language

With v0.6, `\wikiref{<lemma>}{<text>}` works like

```
\wikilangref{<lc>}{<lemma>}{<text>}
```

when `\langcode` expands to `<lc>` (the two-letter language code according to ISO 639-1). The default for `<lc>` is ‘en’ for English. It can be overridden even before loading texlinks (e.g., by an earlier `\newcommand\langcode{de}`):

```

81 \providecommand*{\langcode}{en}

```

For the German versions, use `\renewcommand{\langcode}{de}`. The langcode package provides a command `\uselangcode{<lc>}` that works like `\renewcommand*{\langcode}{<lc>}` and adjusts a number of other settings.

```
82 \newcommand*{\wikiref}{\wikilangref\langcode}
```

`\Wikiref` and `\Wikidisambref` are the obvious analogues:

```
83 \newcommand*{\Wikiref}{\Wikilangref\langcode}
```

```
84 \newcommand*{\Wikidisambref}{\Wikilangdisambref\langcode}
```

## 8.5 Blanks and Umlauts in URLs and Anchors

`\underscorechar` seemed to be useful in macro definitions. The name was inspired by L<sup>A</sup>T<sub>E</sub>X's `\@backslashchar` and `\@percentchar`. However, I am now trying what happens without it. It occurred in `blog.tex` for the documentation of the `blog` package, but `\string_` seems to be a good replacement.

```
85 % \newcommand \underscorechar {}
```

```
86 % {\@makeother\_ \gdef\underscorechar{\_}}
```

Anyway, in my notes I have a more elegant macro for providing “other” versions of special characters.

Guessing what `\underscorechar` was good for (2011-05-17): Wikipedia lemmas and anchors often or even *typically* contain *blank spaces*. The Wikipedia software usually converts them into underscore characters. Blank spaces in *lemmas* seem *not* to need treatment here in `texlinks`. However, Wikipedia also creates *anchors* from *section headings*, which typically contain blank spaces. This has been more difficult ...

Likewise with umlauts: text encoding suffices for *lemmas* (my `\urluml` is not needed for this purpose). But umlauts in *anchors* generated from *section headings* are different. While umlauts in *lemmas* are represented by sequences starting with a *percent* character, the anchors use a *dot* instead of the percent character. Therefore now `\ancuml{<char>}` is provided:

```
87 \newcommand*{\ancuml}[1]{\csname ancuml:#1\endcsname}
```

```
88 \@namedef{ancuml:a}{.C3.A4}
```

```
89 \@namedef{ancuml:o}{.C3.B6}
```

```
90 \@namedef{ancuml:u}{.C3.BC}
```

```
91 \@namedef{ancuml:s}{.C3.9F}
```

```
92 % \newcommand*{\itwikideref}[2]{\wikideref{#1}{\textit{#2}}}
```

```
93 % \newcommand*{\itwikienref}[2]{\wikienref{#1}{\textit{#2}}}
```

```
94 % \newcommand*{\urluml}[1]{\csname urluml:#1\endcsname}
```

```
95 % \@namedef{urluml:a}{\#C3\#A4}
```

```
96 % \@namedef{urluml:o}{\#C3\#B6}
```

```
97 % \@namedef{urluml:u}{\#C3\#BC}
```

```
98 % \@namedef{urluml:s}{\#C3\#9F}          %% 2010/08/09
```

## 9 T<sub>E</sub>X-related

### 9.1 CTAN

#### 9.1.1 Directories and Files in a T<sub>E</sub>X Archive

`\tugctanref{<path>}{<text>}` makes `<text>` a link to a T<sub>E</sub>X Archive directory or file `<path>`:

```
99 \newcommand*{\tugctanref}{\httpbaseref{tug.ctan.org/tex-archive}}
```

Alternatively, you can refer to an (automatically chosen) CTAN *mirror* using

`\mirrorctanref{<path>}{<text>}`.

(I prefer the *appearance* of the TUG archive, designed by Jim Hefferon.)

```
100 \newcommand*{\mirrorctanref}{\httpbaseref{mirror.ctan.org}}
```

You may actually want to “open” a file `<file-name>` in `<path>` on CTAN, `<file-name>` displayed as the link text, either by

`\tugctanfileref{<path>}{<file-name>}`

or (for a mirror) by

`\mirrorctanfileref{<path>}{<file-name>}`.

```
101 \newcommand*{\tugctanfileref}[2]{%
102   \tugctanref{#1/#2}{\filenamefmt{#2}}}
103 \newcommand*{\mirrorctanfileref}[2]{%
104   \mirrorctanref{#1/#2}{\filenamefmt{#2}}}
```

Typically, L<sup>A</sup>T<sub>E</sub>X macro packages in `macros/latex/contrib/` are discussed, so here is `\ltxcontrib` saving a few characters:

```
105 \newcommand*{\ltxcontrib}{macros/latex/contrib/}
```

`\ctanref` works like `\tugctanref` or like `\mirrorctanref`, depending on `\usetugctan` vs. `\usemirrorctan`. So in any case its syntax is

`\ctanref{<path>}{<text>}`.

Likewise, `\ctanfileref` works like

`\tugctanfileref` or `\mirrorctanfileref`,

depending on the same `\usetugctan` vs. `\usemirrorctan`, so the syntax is

`\ctanfileref{<path>}{<file-name>}`

```
106 \newcommand*{\ctanref}{} \newcommand*{\ctanfileref}{}
107 \newcommand*{\usemirrorctan}{%
108   \let \ctanref \mirrorctanref
109   \let \ctanfileref \mirrorctanfileref}
110 \newcommand*{\usetugctan}{%
111   \let \ctanref \tugctanref
112   \let \ctanfileref \tugctanfileref}
```

`\usemirrorctan` is the **default**, i.e., `\ctanref` and `\ctanfileref` use `mirror.ctan.org`:

```
113 \usemirrorctan
```

*Remark (TODO):* Another implementation I consider is using some `\ctanurl-` prefix that you can redefine for accessing your favourite mirror.

### 9.1.2 Jürgen Fenn’s Topical T<sub>E</sub>X Catalogue

`\bytopicref{⟨anchor⟩}{⟨text⟩}` makes `⟨text⟩` a link to `⟨anchor⟩` of Jürgen Fenn’s Topical Index of the T<sub>E</sub>X 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.

```
114 \newcommand*{\bytopicref}[1]{%
115     \mirrorctanref{help/Catalogue/bytopic.html\##1}}
```

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

### 9.1.3 Jim Hefferon’s Package Descriptions

For v0.6 and in the spirit of Sec. 7.3, we introduce an auxiliary

`\ctanorgbaseref{⟨more-path⟩}{⟨text⟩}`:

```
116 \newcommand*{\ctanorgbaseref}{\httpbaseref{ctan.org}}
```

`\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<sub>E</sub>X) 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 `makedoc`—before 2012/04/10!).

```
117 \newcommand*{\CtanPkgRef}[2]{%
118     \ctanorgbaseref{pkg/#1}{\pkgnamefmt{#2}}}
119 \newcommand*{\ctanpkgref}[1]{\CtanPkgRef{#1}{#1}}
```

`\ctanpkgstyref{⟨name⟩}` adds ‘.sty’ to the package name:

```
120 \newcommand*{\ctanpkgstyref}[1]{\CtanPkgRef{#1}{#1.sty}}
```

Jim Hefferon also offers pages listing of all the packages of a given author, accessible by her/his `⟨id⟩`. `\ctanpkgauhref{⟨id⟩}{⟨text⟩}` makes `⟨text⟩` a link to that page:

```
121 \newcommand*{\ctanpkgauhref}[1]{\ctanorgbaseref{author/id/#1}}
```

E.g., see *my* packages.

## 9.2 Mailing Lists

v0.7 relies on package `langcode` for `\enmonthname{<month-number>}` and `\demonthname{<month-number>}`, for tricks with language codes extending those in Section 8.4:

```
122 \RequirePackage{langcode}
```

The next definitions are backbones for generating links to web pages about T<sub>E</sub>X mailing lists. `\TL@piper@parse<year>-<month-number>-<id>` will be used for referring to single postings:

```
123 \def\TL@piper@parse#1-#2-#3/{#1-\enmonthname{#2}/#3}
```

```
\texlistyearmonthref<list-ref>{<2-digits>-<month-no>}
```

will generate `\<list-ref>{<path>}` for linking to the list of postings of the `<month-no>`th month in the year 20<2-digits>:

```
124 \newcommand*{\texlistyearmonthref}[2]{\texlist@yearmonthref#1#2\@nil}
125 \def\texlist@yearmonthref#1#2-#3\@nil{#1{20#2-\enmonthname{#3}}}
```

‘<path>’ will be ‘20<2-digits>-<month>’, and `<month>` will be the *English* name of the `<month-no>`th month of the year.

```
\texlanglistmonthref<month-cmd><list-ref>{<2-digits>-<month-no>}
```

will generate `\<list-ref>{<path>}{<month>}` where `<month>` is determined from `<month-no>` by `<month-cmd>`:

```
126 \newcommand*{\texlanglistmonthref}[3]{\texlanglistm@nthref#1#2#3\@nil}
127 \def\texlanglistm@nthref#1#2#3-#4\@nil{%
128     #2{20#3-\enmonthname{#4}}{#1{#4}}}
```

`\detexlistmonthref<list-ref>{<2-digits>-<month-no>}` now could be used for `<list-ref>{<path>}{<month>}` German `<month>` ...

```
129 \newcommand*{\detexlistmonthref}{\texlanglistmonthref\demonthname}
```

... as could be `\entexlistmonthref<list-ref>{<2-digits>-<month-no>}` for *English* `<month>` ...

```
130 \newcommand*{\entexlistmonthref}{\texlanglistmonthref\enmonthname}
```

With proper use of `langcode` however,

```
\texlistmonthref<list-ref>{<2-digits>-<month-no>}
```

*automatically* chooses between English and German `<month>` (according to intention ...):

```
131 \newcommand*{\texlistmonthref}{\texlanglistmonthref\monthname}
```

### 9.3 CTAN Announcements

`\ctanannref{⟨id⟩}{⟨text⟩}` makes `⟨text⟩` a link to the DANTE web page displaying a CTAN announcement. You find `⟨id⟩` by searching

`https://lists.dante.de/pipermail/ctan-ann/`

and then reading the URL. `⟨id⟩` is composed as

`⟨year⟩-⟨month⟩/⟨6-digits⟩.html`

where `⟨year⟩` consists of 4 digits and `⟨month⟩` is an *English* month name:

```
132 \newcommand*{\ctanannref}[1]{%
133     \httpsref{lists.dante.de/pipermail/ctan-ann/#1}}
```

`\ctanannpref{⟨id-code⟩}{⟨text⟩}` is a variant of `\ctanannref` where in place of `⟨id⟩` you only type the third and fourth digit of the year (`⟨2-digits⟩`), then a ‘-’, then the (arabic) number `⟨month-no⟩` of the month (cf. Section 9.2 so far), then another ‘-’, and then the actual internal identifier `⟨running-no⟩` (a number of six digits preceding ‘.html’ of the URL). I.e., ‘`⟨id-code⟩`’ is ‘`⟨2-digits⟩-⟨month-no⟩-⟨running-no⟩`’.

```
134 \newcommand*{\ctanannpref}[1]{%
135     \ctanannref{TL@piper@parse#1/.html}}
```

`\ctanannyearmonthref{⟨2-digits⟩-⟨month-no⟩}`

generates `\ctanannref{⟨path⟩}` from ‘`⟨2-digits⟩-⟨month-no⟩`’—`⟨path⟩` as in Section 9.2 ...

```
136 \newcommand*{\ctanannyearmonthref}{\texlistyearmonthref\ctanannref}
```

`\ctanannmonthref{⟨2-digits⟩-⟨month-no⟩}`

generates `\ctanannref{⟨path⟩}{⟨month⟩}` where `⟨month⟩` obeys `\langcode` ...

```
137 \newcommand*{\ctanannmonthref}{\texlistmonthref\ctanannref}
```

### 9.4 TUG

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

```
138 \newcommand*{\tugref}{\httpbaseref{tug.org}}
```

### 9.4.1 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. `⟨id⟩` is composed as `⟨year⟩-⟨month⟩/⟨6-digits⟩.html`.

```
139 \newcommand*{\texhaxref}[1]{\tugref{pipermail/texhax/#1}}
```

`\THref{⟨id⟩}` saves you from choosing `⟨text⟩` and uses `texhax` instead.

```
140 \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 the 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.

```
141 \newcommand*{\texhaxpref}[1]{%           %% 2010/09/07
142 \texhaxref{20\TL@piper@parse#1/.html}} %% 2011/05/03
```

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

`\texhaxyearmonthref{⟨2-digits⟩-⟨month-no⟩}`

generates `\texhaxref{⟨path⟩}` from ‘`⟨2-digits⟩-⟨month-no⟩`’—`⟨path⟩` as in Section 9.2 ...

```
143 \newcommand*{\texhaxyearmonthref}{\texlistyearmonthref\texhaxref}
```

`\texhaxmonthref{⟨2-digits⟩-⟨month-no⟩}`

generates `\texhaxref{⟨path⟩}{⟨month⟩}` where `⟨month⟩` obeys `\langcode` ...

```
144 \newcommand*{\texhaxmonthref}{\texlistmonthref\texhaxref}
```

### 9.4.2 Other

`\tugbartref{tb⟨vol⟩-⟨issue⟩/⟨filename-base⟩}{⟨text⟩}` makes `⟨text⟩` a link to the TUGboat article `⟨filename-base⟩.pdf` in vol. `⟨vol⟩` and issue `⟨issue⟩`:

```
145 % \newcommand*{\tugbartref}[1]{\tugref{TUGboat/Articles/#1.pdf}}
146 \newcommand*{\tugbartref}[1]{\tugref{TUGboat/#1.pdf}}
```

`\tugiref{⟨anchor⟩}{⟨text⟩}` makes `⟨text⟩` a link to an `⟨anchor⟩` on the TUG web page entitled ‘TeX Resources on the Web’ (e.g., ‘Web Projects’):

```
147 \newcommand*{\tugiref}[1]{\tugref{interest.html\##1}}
```

It was `\TUGIref` until v0.6, we keep this for compatibility (deprecated):

```
148 \newcommand*{\TUGIref}{} \let\TUGIref\tugiref
```

## 9.5 UK FAQ

`\ukfaqref{<label>}{<text>}` makes `<text>` a link to the UK T<sub>E</sub>X FAQ page with “label” = `<label>`:

```
149 \newcommand*{\ukfaqref}[1]{\httpref{%
150     www.tex.ac.uk/cgi-bin/texfaq2html?label=#1}}
```

## 9.6 Wikibooks

`\wikilangbooksref{<language-code>}{<book>/<subject>}{<text>}`

```
151 \newcommand*{\wikilangbooksref}[1]{%           %% ‘lang’ 2012/01/06
152     \httpbaseref{#1.wikibooks.org/wiki}}
```

`\latexwikibookref{<subject>}{<text>}` refers to the (English) L<sup>A</sup>T<sub>E</sub>X Wiki-book:

```
153 \newcommand*{\latexwikibookref}[1]{\wikilangbooksref{en}{LaTeX/#1}}
```

The German L<sup>A</sup>T<sub>E</sub>X-Kompodium is somewhat difficult, I leave it for now ...

# 10 Leaving and Version HISTORY

```
154 \endinput
```

## VERSION HISTORY

```
155 v0.1    2011/01/24  new file, code from blog.sty v0.3
156 v0.2    2011/01/27  \urlfoot, \NormalHTTTPref, \foothttppurlref,
157                    "outline" adjusted;
158                    more consistent use of \newcommand and
159                    \@ifdefinable (TODO: guarded \let)
160 v0.3    2011/02/10  [blog]; \urlpkgfoot
161 v0.4    2011/04/27  doc. \tugbartref\ corrected
162                    2011/04/30  shortened link in \tugbartref
163                    2011/05/03  \TL@piper@parse, tried \ctanannref
164                    2011/05/13  reworking Wikipedia, arbitrary languages
165                    2011/06/27  doc.: \acro; \httpsref, \ctanannref
166                    2011/07/23  doc.: typo \acro{TUG}, ‘Almost all’, page breaks;
167                    \Wikidisambref: different order of arg.s
168                    2011/08/18  doc.: \acro with UK; wikibooks
169                    2011/08/27  doc. \acro with URL and PDF;
170                    more doc and code changes for https
171  uploaded with MOREHYPE r0.4 (not touched by r0.41)
172 v0.41    2011/09/03  doc.: more specific on \urluml/Wikipedia
173                    2011/10/06  \mirrorctanref, \tugctanfileref,
174                    \mirrorctanfileref, \ltxcontrib
175                    2011/10/10  doc. formatting of previous
176  uploaded with MOREHYPE r0.5(1)
177 v0.5    2011/10/19  doc. fix LaTeX Wikibook
```



```

178          2011/10/20 \urlfmt, \filenamefmt and \pkgnamefmt
179                    changed and moved, modified doc. on them,
180                    doc. uses \URL
181          2011/10/21 re-order CTAN, \pagebreak's, \ctanref and
182                    choice for it, doc. modified; rm. \ithttpref
183  uploaded with MOREHYPE r0.52
184  v0.6    2012/01/06 \wikilangref etc., \wikiref etc. depend on
185                    \langcode
186          2012/01/11 removed old comments for Wikipedia; (C)
187          2012/03/09 "URL bases" (\httpbaseref etc.), applied;
188                    \bytopicref uses \mirrorctanref
189          2012/03/12 fixed \texhaxref
190          2012/04/09 \ctanorgbaseref, \ctanpkggauref
191          2012/04/10 makedoc link works!
192          2012/05/13 example for \wikilangref corrected
193  uploaded with MOREHYPE r0.6
194  v0.7    2012/07/23 doc.: <text>
195          2012/08/05 \tugiref
196          2012/10/04 doc. wikibooks: ref
197          2012/10/24 ...monthref... requiring 'langcode.sty', moving
198                    links to mailing list pages from 'texblog.fdf'
199                    here; doc.: \pagebreak s, wikibooks: <book>...,
200                    corr. args, \wikiref refers to 'langcode.sty',
201                    'ref', using \qtdcode (new in 'makedoc.cfg')
202          2012/11/08 doc.: Jim corr.
203          2012/11/27 \ctanpkgstyref from 'texblog.fdf'
204          2012/11/28 [[...|...]]
205

```