The `acroterm` package

Jakob Voss

2010/11/19 v0.1

Contents

1 Introduction

There are several packages to manage acronyms in TEX. CTAN lists at least `acronym`,[1] `acromake`,[2], `glosstex`,[3] and `glossary`.[4] These packages let you define acronyms, that can be expanded automatically. But you rarely control when to expand, and defining acronyms can be annoying. The `acroterm` package lets you naturally use acronyms and technical terms in you document. You can mark up acronyms (`\acro`) and terms (`\term`) as such, to get them formatted consistently and to include them in an index. In addition you can connect acronyms and terms that belong together. You fully control, where a term is used in full form, and where it is used as acronym. Tracking the first ocurrence of a term is not the primary goal of this package, but may implemented in a later version.

The current version of this package is a developer version that still depends on the packages `splitidx` and `hyperref`. Please load `acroterm` between the two.

A small example is included in this package in file `example.tex`.

2 Description

Essentially, this package provides four macros to mark terms, acronyms, and both:

---

[1] http://www.ctan.org/tex-archive/macros/latex/contrib/acronym/

[2] http://www.ctan.org/tex-archive/macros/latex/contrib/acromake/

[3] http://www.ctan.org/tex-archive/support/glosstex/

[4] http://www.ctan.org/tex-archive/macros/latex/contrib/glossary/

`\acro{SNAFU}` ⟶ SNAFU
`\term{Potrzebie}` ⟶ Potrzebie
`\tacro{Do it yourself}{DIY}` ⟶ Do it yourself (DIY)
`\aterm{DIY}{Do it yourself}` ⟶ DIY (Do it yourself)

And, after definition of an acronym like above:

`\acroexpand{DIY}` ⟶ Do it yourself

The macros `\term` and `\acro` simply set some text as term or as acronym. `\tacro` and `\aterm` mark a term together with its acronym or vice versa. In all cases, terms and acronyms are put into a special index that can be used later. Layout can be controlled globally.

`\term` and `\tacro` support an optional parameter for indexing. With

```
\term[Potrzebie System of Weights and Measures]{Potrzebie System}
```

the term ‘Potrzebie System of Weights and Measures’ is included in the index but ‘Potrzebie System‘ is printed in the text. In the same way

```
\tacro[Gang of Four (Patterns)]{Gang of Four}{GoF}
```

prints ‘Gang of Four (GOF)‘ but indexes ‘Gang of Four (Patterns)’.

Each command has an emphasizing variant that starts with an uppercase letter:

`\Acro{SNAFU}` ⟶ SNAFU
`\Term{Potrzebie}` ⟶ *Potrzebie*
`\Tacro{Do it yourself}{DIY}` ⟶ *Do it yourself* (DIY)
`\Aterm{DIY}{Do it yourself}` ⟶ DIY (Do it yourself)

3 Implementation

The current version of `acroterm` depends on `splitidx` for index generation. This dependency may be removed in a future version.

```
\RequirePackage{splitidx}[2009/02/18 v1.1a]
\RequirePackage{xifthen}
```

Styles

The following commands are used to simply print acronyms and terms. They do not index but only format the arguments. You can redefine them to change layout of acronyms and terms. The uppercase variant is used for emphasizing.

`\acrostyle` print an acronym in normal form

`\Acrostyle` print an acronym in emphasized form

```
\newcommand{\Acrostyle}[1]{#1}
\newcommand{\acrostyle}[1]{\textsc{\lowercase{#1}}}
```

`\termstyle` print a term in normal form

`\Termstyle` print a term in emphasized form

```
\newcommand{\termstyle}[1]{#1}
\newcommand{\Termstyle}[1]{\textit{#1}}
```

`\tacrostyle` print a term and its acronym in normal form

`\Tacrostyle` print a term and its acronym in emphasized form

```
\newcommand{\tacrostyle}[2]{\termstyle{#1} (\acrostyle{#2})}
\newcommand{\Tacrostyle}[2]{\Termstyle{#1} (\acrostyle{#2})}
```

`\atermstyle` print an acronym and its term in normal form

`\Atermstyle` print an acronym and its term in emphasized form

```
\newcommand{\atermstyle}[2]{\acrostyle{#1} (\termstyle{#2})}
\newcommand{\Atermstyle}[2]{\acrostyle{#1} (\termstyle{#2})}
```

`\acroindexstyle` print an acronym as index entry.

```
\newcommand{\acroindexstyle}[1]{#1\acroexpand[ --- ]{#1}}
```

Internal macros

`\provideacronym` Connects an acronym (that must only contain letters) with a term. No output or index entry is generated. In most cases you do not need to explicitly use this macro.

```
\newcommand{\provideacronym}[2]{% TODO: detect redefinitions
  \expandafter\providecommand\expandafter{%
  \csname acronymlong#1\endcsname}{#2}%
}
```

Index generation

The following hack is required hack to mix hyperref and formatted page numbers. It may be changed because `\bfhref` may already have been defined.

```
\newcommand{\bfhref}[1]{\textbf{\hyperpage{#1}}}
```

And some code for index generation (also to be fixed).

```
\newcommand{\acro@define}[2]{% #1: long term, #2: acronym
  \sindex[idx]{#1|see{\acrostyle{#2}}}% TODO: acronyms in general index?
  \@ifundefined{acronymlong#2}{%
    \provideacronym{#2}{#1}}{}%
  \sindex[acronym]{#2@\acroindexstyle{#2}|bfhref}%
}
```

`\acroexpand` print the expanded form (that is the term) of an acronym, if defined. The optional first parameter is prepended only on success.

```
\newcommand{\acroexpand}[2][]{%
  \@ifundefined{acronymlong#2}{}{%
  #1\csname acronymlong#2\endcsname}%
}
```

Main macros

`\term` marks a term in normal form. Parameters: `[INDEX]{TERM}`

```
\newcommand{\term}[2][]{%
  \ifthenelse{\isempty{#1}}%
  {\sindex[idx]{#2}}{\sindex[idx]{#1}}%
  \termstyle{#2}}
```

`\Term` marks a term in emphasized form. Parameters: `[INDEX]{TERM}`

```
\newcommand\Term[2][]{%
  \ifthenelse{\isempty{#1}}%
  {\sindex[idx]{#2|bfhref}}%
  {\sindex[idx]{#1|bfhref}}%
  \Termstyle{#2}%
}
```

`\acro` marks an acronym in normal form. Parameters: `{ACRO}`

```
\newcommand{\acro}[1]{%
  \acrostyle{#1}%
  {\sindex[acronym]{#1@\acroindexstyle{#1}}}%
}
```

`\Acro` marks an acronym in emphasized form. Parameters: `{ACRO}`

```
\newcommand{\Acro}[1]{%
  \Acrostyle{#1}%
  {\sindex[acronym]{#1@\acroindexstyle{#1}}}%
}
```

`\tacro` Parameters: `[INDEX]{TERM}{ACRO}`

```
\newcommand{\tacro}[3][]{%
  \ifthenelse{\isempty{#1}}%
  {\acro@define{#2}{#3}}{\acro@define{#1}{#3}}% TODO: not define?
  \tacrostyle{#2}{#3}}
```

`\Tacro` Parameters: `[INDEX]{TERM}{ACRO}`

```
\newcommand{\Tacro}[3][]{%
  \ifthenelse{\isempty{#1}}%
  {\acro@define{#2}{#3}}{\acro@define{#1}{#3}}%
  \Tacrostyle{#2}{#3}}
```

`\aterm` Parameters: `{ACRO}{TERM}`

```
\newcommand{\aterm}[2]{%
  \acro@define{#2}{#1}% TODO: not define but only use?
  \atermstyle{#1}{#2}}
```

`\Aterm` Parameters: `{ACRO}{TERM}`

```
\newcommand{\Aterm}[2]{%
  \acro@define{#2}{#1}%
  \Atermstyle{#1}{#2}}
```

Bugs and missing features

Index generation may need to be fixed. We also want to discover multiple definitions of an acronom with different terms. Should terms, that are used without acronym, occurr in the acronym index? Version 1.0 sure needs to support other index generators.

See the section on 'Auxiliary macros for name indexing directives' in biblatex source code for hints on including macros in the index.

The latest development version of this package can be found at `https://github.com/nichtich/acroterm`.