

ACRO

VO.5 2012/11/28

Typeset Acronyms

Clemens NIEDERBERGER

<https://bitbucket.org/cgnieder/acro/>
contact@mychemistry.eu

English documentation

ACRO not only allows you to create acronyms in a simple way but also lets you add them to different classes of acronyms. Lists can be created of separate classes wherever you want the list to appear.

ACRO also provides an option **single** which ignores acronyms that are used only once in the whole document.

As an experimental feature **ACRO** also offers the option **sort** which automatically sorts the list created by `\printacronyms`.

Contents

	4.3. Labels	7
	4.4. <code>\ac</code> and Friends in PDF Bookmarks	7
1. Licence and Requirements	2	
2. About	2	
3. Basics	2	
3.1. Creating New Acronyms . . .	2	
3.2. Using the Acronyms – the Commands	3	
3.3. Plural Forms	5	
3.4. Alternative Short Forms . . .	5	
3.5. Extra Information for the List Entry	6	
4. Additional Commands and Possibilities	6	
4.1. Using Classes	6	
4.2. Reset or Mark as Used	7	
	5. Printing the List	8
	6. Customization	9
	6.1. General Options	9
	6.2. Options Regarding Acronyms	11
	6.3. Options Regarding the List . .	11
	7. hyperref Support	13
	A. All Acronyms Used in this Documentation	13
	References	14
	Index	16

1. Licence and Requirements

Permission is granted to copy, distribute and/or modify this software under the terms of the L^AT_EX Project Public License, version 1.3 or later (<http://www.latex-project.org/lppl.txt>). The package has the status “maintained.”

ACRO loads and needs the following packages: `expl3`,¹ `xparse`,² `xtemplate`³ and `l3keys2e`.⁴

2. About

Acronyms and initialisms are abbreviations formed from the initial components in a phrase or a word. These components may be individual letters (as in CEO) or parts of words (as in Benelux and Ameslan). There is no universal agreement on the precise definition of the various terms nor on written usage. [Wik12a]

After WIKIPEDIA told us what acronyms are and we won’t confuse them with units or other kinds of abbreviations – why would we need another package for them? There are several already: `acronym`⁵ [Oet10], `acromake`⁶ [Tho95], `acroterm`⁷ [Vos10], the abbreviations package `abbrevs`⁸ [Swio1] (the current version 1.4 has a bug,⁹ though), the nomenclature package `nomenc1`¹⁰ [Vey+05], and of course the mighty glossaries¹¹ [Tal11]. So there is really no *need* for a new package.

On the other hand `acronym`, the best of the acronym specific packages, has one or two shortcomings and sometimes using glossaries seems a bit of an overkill (or simply inconvenient as one has to run `makeglossaries`, `makeindex` or `xindy`, then). So **ACRO** stands somewhere in between (but much closer to `acronym`).

The main reason for the existence of **ACRO** is a question on `{TeX}.SE`¹² which intrigued me and in consequence led to **ACRO** and it’s option `single`.

ACRO has many similarities with the `acronym` package. In fact, quite some macros have the same name and meaning.¹³

3. Basics

3.1. Creating New Acronyms

Acronyms are created with the command `\DeclareAcronym` that can only be used in the preamble.

`\DeclareAcronym*{<id>}{<short>,<pl>}[<alt.>]{<long>,<pl>}{<extra>}[<class>]` % can be used only in preamble

`\DeclareAcronymFormat{<id>}{<format>}` % can be used only in preamble

`\DeclareAcronymCitation{<id>}[<pre>][<post>]{<cite keys>}` % can be used only in preamble

¹ CTAN: `expl3` ² CTAN: `xparse` ³ CTAN: `xtemplate` ⁴ CTAN: `l3keys2e` ⁵ CTAN: `acronym` ⁶ CTAN: `acromake` ⁷ CTAN: `acroterm` ⁸ CTAN: `abbrevs` ⁹ see <http://tex.stackexchange.com/q/59840/5049> for solutions. ¹⁰ CTAN: `nomenc1` ¹¹ CTAN: `glossaries` ¹² <http://tex.stackexchange.com/q/59449/5049>

¹³ Not in the sense of `\meaning!`

`\DeclareAcronymPDFString{<id>}{<pdf entry>,<plural ending>}` % can be used only in preamble

Although `\DeclareAcronym` has quite a number of arguments. In the simplest case you'll only need three:

```
1 % preamble
2 \DeclareAcronym{test}{ST}{Some Test}
```

This creates the acronym “ST” with the ID “test” and the long form “Some Test.” The other arguments will be described later in this document.

`\DeclareAcronymFormat` lets you assign a special format to an acronym. Maybe small caps:

```
1 % preamble:
2 \DeclareAcronym{ot}{ot}{Other Test}
3 \DeclareAcronymFormat{ot}{\scshape}
```

The short form now looks like this: `ot`.

With `\DeclareAcronymCitation` you can add a key as used in `\cite` that is added to the first or every appearance of the acronym (depending on an option that is described later).

```
1 % preamble:
2 \DeclareAcronym{ny}{NY,}{New York,}
3 \DeclareAcronymCitation{ny}{NewYork}
4 % bib file for use with biber/biblatex:
5 @online{NewYork,
6   author = {Wikipedia},
7   title  = {New York City},
8   urldate = {2012-09-27},
9   url    = {http://en.wikipedia.org/wiki/New_York_City},
10  year   = {2012}
11 }
```

The first appearance now looks as follows¹⁴: New York (NY) [Wik12b].

3.2. Using the Acronyms – the Commands

Acronyms are used with with one of the following commands:

`\ac*{<id>}` → basic command; the first output is different from subsequent ones.

`\Ac*{<id>}` → same as `\ac` but capitalizes the first letter of the long form.

¹⁴ The appearance of the citation of course depends on the citation style you're using.

`\acs*{<id>}` → **s**hort form; the actual acronym.

`\acl*{<id>}` → **l**ong form; the meaning of the acronym.

`\Acl*{<id>}` → same as `\acl` but capitalizes first letter.

`\aca*{<id>}` → **a**lternative short form as specified in the [`<short t>`] argument of `\DeclareAcronym`; if it hasn't been specified this is identical to `\acs`.

`\acf*{<id>}` → first form; output like the first time `\ac` is output.

`\Acf*{<id>}` → same as `\acf` but capitalizes first letter of the long form.

`\acp*{<id>}` → **p**lural form of `\ac`;

`\Acp*{<id>}` → same as `\acp` but capitalizes first letter of the long form.

`\acsp*{<id>}` → plural form of `\acs`;

`\aclp*{<id>}` → plural form of `\acl`;

`\Aclp*{<id>}` → same as `\aclp` but capitalizes first letter.

`\acap*{<id>}` → plural form of `\aca`;

`\acfp*{<id>}` → plural form of `\acf`;

`\Acfp*{<id>}` → same as `\acfp` but capitalizes first letter of the long form.

If an acronym is used the first time with `\ac` its output is different from subsequent uses. To be clear on this: the first time! If the acronym has been used with *any* of the output commands before it is *not* the first time any more.

If you use the starred variant an acronym will not be marked as used. This proves useful if an acronym is typeset in a section title, for example, since then the appearance in the table of contents won't mark it as used.

Introduced in
version 0.5

```

1 % preamble: \DeclareAcronym{cd}{cd}{Compact Disc}
2 % \DeclareAcronymFormat{cd}{\scshape}
3 first time: \ac{cd} \\
4 second time: \ac{cd} \\
5 short: \acs{cd} \\
6 alternative: \aca{cd} \\
7 first again: \acf{cd} \\
8 long: \acl{cd} \\
9 short plural: \acsp{cd} \\
10 long plural: \aclp{cd}

```

first time: Compact Disc (CD)
 second time: CD
 short: CD
 alternative: CD
 first again: Compact Disc (CD)
 long: Compact Disc
 short plural: CDs
 long plural: Compact Discs

3.3. Plural Forms

If an acronym is defined in the standard way `\ACRO` uses an ‘s’ that’s appended to both the short and the long form when one of the plural commands is used. However, that is not always the best solution. For one thing not all acronyms may have a plural form. Second, the plural form especially of the long forms may be formed differently. And third, other languages can have other plural endings.

For this reason `\DeclareAcronym` provides additional arguments:

```
\DeclareAcronym{<id>}{<short>,<plural ending>}{<long>,<plural ending>}
```

```
\DeclareAcronym*{<id>}{<short>,<plural ending>}{<long>,<plural form>}
```

These arguments are optional. If they’re not used, the default setting is ‘s’. If you use the starred version the long form will be replaced by the specified plural form when necessary. A given short plural will still be appended.

```

1  % preamble:
2  % \DeclareAcronym{cd}{\textsc{cd}}{Compact Disc}
3  % \DeclareAcronymFormat{cd}{\scshape}
4  % \DeclareAcronym{ny}{NY,}{New York,}
5  % \DeclareAcronym{sw}{SW}{Sammelwerk,e}
6  % \DeclareAcronym*{mp}{MP}{Member of Parliament,Members of Parliament}
7  \acsp{cd}, \aclp{cd} \\\
8  \acsp{ny}, \aclp{ny} \\\
9  \acsp{sw}, \aclp{sw} \\\
10 \acsp{mp}, \aclp{mp}

CDs, Compact Discs
NY, New York
SWs, Sammelwerke
MPs, Members of Parliament

```

3.4. Alternative Short Forms

For some acronyms it might be useful to have alternative forms. For this `\DeclareAcronym` has another optional argument:

```
\DeclareAcronym{<id>}{<short>}[<alternative>]{<long>}
```

```

1 % preamble \DeclareAcronym{jpg}{JPEG}[JPG]{Joint Photographic Experts
  Group}
2 default: \acs{jpg} \
3 alt.: \aca{jpg}

default: JPEG
alt.: JPG

```

The alternative form uses the same plural ending as the default short form and is formatted in the same way.

3.5. Extra Information for the List Entry

Of course you can print a list of acronyms where their meaning is explained. Sometimes it can be useful to add additional information there. This is done with another optional argument to `\DeclareAcronym`:

```
\DeclareAcronym{<id>}{<short>}{<long>}{<extra>}
```

These information will only be displayed in the list. See section 5 for the impact of the following example.

```

1 % preamble:
2 % \DeclareAcronym{nato}{nato}{North Atlantic Treaty Organization}{\textit{
  deutsch}: Organisation des Nordatlantikvertrags}
3 % \DeclareAcronymFormat{nato}{\scshape}
4 The \ac{nato} is an intergovernmental military alliance based on the
5 North Atlantic Treaty which was signed on 4~April 1949. \ac{nato}
6 headquarters are in Brussels, Belgium, one of the 28 member states
7 across North America and Europe, the newest of which, Albania and
8 Croatia, joined in April 2009.

```

The North Atlantic Treaty Organization (NATO) is an intergovernmental military alliance based on the North Atlantic Treaty which was signed on 4 April 1949. NATO headquarters are in Brussels, Belgium, one of the 28 member states across North America and Europe, the newest of which, Albania and Croatia, joined in April 2009.

4. Additional Commands and Possibilities

4.1. Using Classes

The acronyms of `ACRO` can be divided into different classes. This doesn't change the output but allows different acronym lists, see section 5. For this `\DeclareAcronym` has an optional argument:

```
\DeclareAcronym{<id>}{<short>}{<long>}[<class>]
```

```

1 % preamble:
2 % \DeclareAcronym{la}{LA,}{Los Angeles,}[city]
3 % \DeclareAcronym{ny}{NY,}{New York,}[city]
4 \acl{la} (\acs{la}) \l
5 \acl{ny} (\acs{ny})

Los Angeles (LA)
New York (NY)

```

This might be useful if you can and want to divide your acronyms into different types, technical and grammatical ones, say, that shall be listed in different lists.

4.2. Reset or Mark as Used

If you want for some reason to fool **ACRO** into thinking that an acronym is used for the first time you can call one of these commands:

Introduced in version 0.5 `\acreset{<comma separated list of ids>}`

`\acresetall`

```

1 \acreset{ny}\ac{ny}
New York (NY) [Wik12b]

```

Beware that both commands act *globally*! There is also one command that effectively is the opposite of `\acreset`, i.e., it marks an acronym as used:

Introduced in version 0.5 `\acuse{<comma separated list of ids>}`

4.3. Labels

The first usage of an acronym sets a label `\label{ac:<id>}`, regardless which command has been used to typeset it. This label is used to create the page numbers in the list (see section 5 for details) but can also be used for other references if needed.

4.4. \ac and Friends in PDF Bookmarks

Introduced in version 0.5 **ACRO**'s commands usually are not expandable which means they'd leave unallowed tokens in PDF bookmarks. `hyperref`¹⁵ offers `\texorpdfstring` to circumvent that issue manually but that isn't really a nice solution. What's the point of having macros to get output for you if you have to specify it manually after all?

That is why **ACRO** offers a preliminary solution for this. In a bookmark every `\ac` like command falls back to a simple text string typesetting what `\acs` would do (or `\acsp` for plural forms). These text strings both can be accessed manually and can be modified to an output reserved for PDF bookmarks.

¹⁵ CTAN: hyperref

`\acpdfstring{<id>}` → access the text string used in PDF bookmarks.

`\acpdfstringplural{<id>}` → access the plural form of the text string used in PDF bookmarks.

`\DeclareAcronymPDFString{<id>}{<pdf entry>,<plural ending>}` → declare a custom text string for PDF bookmarks.

For example the PDF acronym used in the title for this section is defined as follows:

```
1 \DeclareAcronym{pdf}{pdf}{Portable Document File}
2 \DeclareAcronymFormat{pdf}{\scshape}
3 \DeclareAcronymPDFString{pdf}{PDF}
```

5. Printing the List

Printing the whole list of acronyms is easy: just place `\printacronyms` wherever you want the list to be.

`\printacronyms{<list of classes>}[<excluded classes>]` → the first *optional* (!) argument {<list of classes>} takes a comma-separated list of the classes of acronyms that should be in the list. If the argument is left out, *all* acronyms are printed. In the optional argument [<excluded classes>] classes can be specified that shall not be included into the list.

```
1 \acsetup{hyperref=false,extra-style=comma}
2 \printacronyms[city]
3 \acsetup{list-name=City Acronyms}
4 \printacronyms{city}
```


Acronyms

CD Compact Disc

ID identification string

JPEG Joint Photographic Experts Group

MP Member of Parliament

NATO North Atlantic Treaty Organization, *deutsch*: Organisation des Nordatlantikvertrags

PDF Portable Document Format

SW Sammelwerk

ST Some Test

City

Acronyms

LA Los Angeles

NY New York

You can see that the default layout is a description list with a section* title. Both can be changed, see section 6.

The command `\printacronyms` needs two \LaTeX runs. This is a precaution to avoid error messages with a possibly empty list. But since almost all documents need at least two runs and often are compiled much more often than that, this fact shouldn't cause too much inconvenience.

6. Customization

6.1. General Options

There are a few options which change the general behaviour of **ACRO**. Underlined values are used if no value is given.

`single` = true|false Default: false

If set to true an acronym that's used only once (with `\ac`) in a document will only print the long form and will not be printed in the list.

`hyperref` = true|false Default: false

If set to true the short forms of the acronyms will be linked to their list entry.

`only-used` = true|false Default: true

This option is true as default. It means that only acronyms that are actually used in the document are printed in the list. If false, all acronyms defined with `\DeclareAcronym` will be written to the list.

`macros = true|false` Default: false
If set to true this option will create a macro `\<id>` for each acronym as a shortcut for `\ac{<id>}`. Already existing macros will *not* be overwritten.

`strict = true|false` Default: false
If set to true and the option `macros = true` is in effect then already existing macros will be overwritten.

`sort = true|false` Default: false
If set to true the acronym list will be sorted automatically. The entries are sorted by their ID ignoring upper and lower case. This option needs the experimental package `l3sort` (from the `l3experimental`¹⁶ bundle) and can only be set in the preamble.

`cite = all|first|none` Default: first
This option decides whether citations that are added via `\DeclareAcronymCitation` are added to each first, every or no appearance of an acronym.

`cite-cmd = <cmd>` Default: `\cite`
This option determines which command is used for the citation. Each citation command that takes the cite key as argument is valid, for example `biblatex`'s¹⁷ `\footcite`.

`cite-space = <code>` Default: `\nobreakspace`
Depending on the citation command in use a space should be inserted before the citation or maybe not (e.g. `\footcite...`). This option allows you to set this. Actually it can be used to place arbitrary code right before the citation.

`uc-cmd = <cmd>` Default: `\acro_first_upper_case:n`
The command that is used to capitalize the first word in the `\Ac` and the like commands. You can change it to another one like for example `\makefirstuc`¹⁸ or `\MakeTextUppercase`.¹⁹

All options of this and the following sections can be set up either as package options or via the `setup` command:

`\acsetup{<options>}` set up `ACRO` anywhere in the document. Or separate package loading from `setup`.

```
1 % with \acsetup{macros}
2 we could have used these before: \nato, \ny

we could have used these before: NATO, NY
```

¹⁶ CTAN: l3experimental ¹⁷ CTAN: biblatex ¹⁸ from the mfirstuc package ¹⁹ from the textcase package

6.2. Options Regarding Acronyms

The options described in this section all influence the layout of one of the possible output forms of the acronyms.

`short-format` = <format> (initially empty)

Sets a format for all short forms. For example `short-format = \scshape` would print all short forms in small caps.

`long-format` = <format> (initially empty)

The same for the long forms.

`list-long-format` = <format> (initially empty)

An extra format for the long entries in the list. If not used this is the same as `long-format`.

`extra-format` = <format> (initially empty)

The same for the extra information.

`first-style` = default|plain|footnote|square|short|reversed|plain-reversed
Default: default

The basic style of the first appearance of an acronym.

`extra-style` = plain|comma|paren|bracket Default: plain

Defines the way the extra information is printed in the list.

`plural-ending` = <tokenlist> Default: s

With this option the default plural ending can be set.

```
1 % (Keep in mind that we're in
2 % a minipage here!)
3 \acsetup{first-style=footnote}
4 \acf{ny} \
5 \acsetup{first-style=square}
6 \acf{ny} \
7 \acsetup{first-style=short}
8 \acf{ny} \
9 \acsetup{first-style=reversed}
10 \acf{ny} \
11 \acsetup{first-style=plain}
12 \acf{ny} \
13 \acsetup{first-style=plain-reversed}
14 \acf{ny}
```

NY^a
New York [NY] [Wik12b]
NY [Wik12b]
NY (New York) [Wik12b]
New York – NY [Wik12b]
NY – New York [Wik12b]

 ^a New York [Wik12b]

6.3. Options Regarding the List

`page-ref` = none|plain|comma|paren Default: none

The style with which the page number of the first appearance of the acronym in the text is displayed.

`page-name` = <page name> Default: p.
The “name” of the page label.

`list-type` = table|<list> Default: description
This option let’s you choose how the list is printed. <list> can be any valid list like itemize or description.

`list-style` = list|tabular|longtable|extra-tabular|extra-longtable|
extra-tabular-rev|extra-longtable-rev Default: list
If you choose `list-type` = table you have to specify which kind of table should be used. If you choose longtable, extra-longtable or extra-longtable-rev you have to load longtable²⁰ in your preamble. The values extra-longtable and extra-longtable-rev put the extra information in a column of its own.

`list-header` = chapter|chapter*|section|section*|subsection|subsection*
The header type of the list. Default: section*

`list-name` = <list name> Default: Acronyms
The name of the list. This is what’s written in the list-header.

`list-table-width` Default: .7\linewidth
This has only an effect if you chose `list-type` = table. The second column (or the third if you choose one of the -rev styles) of the table is a p column whose width can be specified with this option.

`list-caps` = true|false Default: false
Print the first letters of the long form capitalized.

If you for example have loaded enumitem²¹ you can define a custom list for the acronym list:

```
1 % preamble:
2 % \usepackage{enumitem}
3 \newlist{acronyms}{description}{1}
4 \newcommand*{\addcolon[1]{#1:}
5 \setlist[acronyms]{labelwidth=3em,leftmargin=3.5em,noitemsep,itemindent=0
  pt,font=\addcolon}
6 \acsetup{list-type=acronyms,hyperref=false,extra-style=comma}
7 \printacronyms
```

²⁰ CTAN: longtable ²¹ CTAN: enumitem

Acronyms

CD:	Compact Disc
ID:	identification string
JPEG:	Joint Photographic Experts Group
LA:	Los Angeles
MP:	Member of Parliament
NATO:	North Atlantic Treaty Organization, <i>deutsch:</i> Organisation des Nordatlantikvertrags
NY:	New York
PDF:	Portable Document Format
SW:	Sammelwerk
ST:	Some Test

7. hyperref Support

The option `hyperref = true` adds internal links from all short (or alternative) forms to their respective list entries. Of course this only works if you have loaded the `hyperref` package in your preamble. You should use this option with care: if you don't use `\printacronyms` anywhere this option will result in loads of `hyperref` warnings. Also printing several lists can result in warnings if don't clearly separate the lists into different classes. If an acronym appears in more than one list there will also be more than one `hypertarget` for this acronym.

A. All Acronyms Used in this Documentation

```
1 \acsetup{
2   list-type   = table,
3   list-style  = longtable,
4   list-header = subsection*,
5   extra-style = comma,
6   page-ref    = comma,
7   list-name   = All Acronyms}
8 \printacronyms
```

All Acronyms

CD	Compact Disc, p. 5
ID	identification string, p. 3
JPEG	Joint Photographic Experts Group, p. 6
LA	Los Angeles, p. 7
MP	Member of Parliament, p. 5

NATO	North Atlantic Treaty Organization, <i>deutsch</i> : Organisation des Nordatlantikvertrags, p. 6
NY	New York, p. 3
PDF	Portable Document Format, p. 7
SW	Sammelwerk, p. 5
ST	Some Test, p. 3

```

1 \acsetup{
2   list-type   = table,
3   list-style  = longtable,
4   list-header = subsection*,
5   page-ref    = comma,
6   list-name   = City-Acronyms}
7 \printacronyms{city}

```

City-Acronyms

LA	Los Angeles, p. 7
NY	New York, p. 3

References

- [Oet10] Tobias Oetiker. acronym – An Acronym Environment for \LaTeX 2 ϵ . Version 1.36, Sept. 8, 2010. URL: <http://www.ctan.org/pkg/acronym>.
- [Swi01] Matt Swift. abbrevs. Version 1.4, Sept. 8, 2001. URL: <http://www.ctan.org/pkg/abbrevs>.
- [Tal11] Nicola L.C. Talbot. glossaries. Version 3.01, Apr. 12, 2011. URL: <http://www.ctan.org/pkg/glossaries>.
- [Tho95] Paul A. Thompson. acromake. Version na, July 16, 1995. URL: <http://www.ctan.org/pkg/acromake>.
- [Vey+05] Boris Veytsman, Bernd Schandl, Lee Netherton, and C.V. Radhakrishnan. nomencl – A Package to Create a Nomenclature. Version 4.2, Sept. 22, 2005. URL: <http://www.ctan.org/pkg/nomencl>.
- [Vos10] Jakob Voss. acroterm. Version 0.1, Nov. 19, 2010. URL: <http://www.ctan.org/pkg/acroterm>.
- [Wik12a] Wikipedia. *Acronym and initialism*. 2012. URL: <http://en.wikipedia.org/wiki/Acronyms> (visited on 06/21/2012).

[Wik12b] Wikipedia. *New York City*. 2012. URL: http://en.wikipedia.org/wiki/New_York_City (visited on 09/27/2012).

Index

Section titles are indicated **bold**, packages sans serif, commands \brown and options yellow.

Symbols

\<id> 10

A

abbrevs 2
\Ac 3, 10
\ac 1, 3 f., 7, 9 f.
\aca 4
\acap 4
\Acf 4
\acf 4
\Acfp 4
\acfp 4
\Acl 4
\acl 4
\Aclp 4
\aclp 4
\Acp 4
\acp 4
\acpdfstring 8
\acpdfstringplural 8
\acreset 7
\acresetall 7
acromake 2
acronym 2
acroterm 2
\acs 4, 7
\acsetup 10
\acsp 4, 7
\acuse 7

Additional Commands

6 ff.

classes 6
labels 7
resetting 7

B

Basics 2–6
alternative forms 5
extra information 6
new acronyms 2
plural forms 5

using acronyms 3

biblatex 10

C

cite 10
cite-cmd 10
cite-space 10
Customization 9–13
acronyms 11
general options 9
list 11

D

\DeclareAcronym 2–6, 9
\DeclareAcronymCitation
2 f., 10
\DeclareAcronymFormat
2 f.
\DeclareAcronymPDFString
3, 8

E

enumitem 12
expl3 2
extra-format 11
extra-style 11

F

first-style 11

G

glossaries 2

H

hyperref 9, 13
hyperref 7, 13
hyperref support 13

L

l3experimental 10
l3keys2e 2
l3sort 10
\linewidth 12
list-caps 12

list-header 12
list-long-format 11
list-name 12
list-style 12
list-table-width 12
list-type 12
long-format 11
longtable 12

M

macros 10
mfirstuc 10

N

nomencl 2

O

only-used 9

P

page-name 12
page-ref 11
plural-ending 11
\printacronyms 1, 8 f., 13
Printing the List 8 f.

S

\scshape 11
short-format 11
single 1 f., 9
sort 1, 10
strict 10

T

\texorpdfstring 7
textcase 10

U

uc-cmd 10

X

xparse 2
xtemplate 2