1 Introduction

When writing a paper on cellular mobile radio I started to use a lot of acronyms. This can be very disturbing for the reader, as he might not know all the used acronyms. To help the reader I kept a list of all the acronyms at the end of my paper.

This package makes sure, that all acronyms used in the text are spelled out in full at least once.

2 The user interface

The package provides several commands and one environment for dealing with acronyms. Their appearance can be controlled by two package options and three macros.

2.1 Acronyms in the Text

\ac\ To enter an acronym inside the text, use the
\ac{acronym}
command. The first time you use an acronym, the full name of the acronym along with the acronym in brackets will be printed. If you specify the footnote option while loading the package, the full name of the acronym is printed as a footnote. The next time you access the acronym only the acronym will be printed.

\acf\ If later in the text again the Full Name of the acronym should be printed, use the command
\acf{acronym}

\acs\ To get the short version of the acronym, use the command
\acs{acronym}

\acl\ Gives you the expanded acronym without even mentioning the acronym.
\acl{acronym}
\acs{ac} Works in the same way as \ac, but makes the short and/or long forms into English plurals by adding an ‘s’.
\acsp{ac} Works in the same way as \acs, but makes the short form into an English plural by adding an ‘s’.
\acfp{ac} Works in the same way as \acf, but makes the short and long forms into English plurals by adding an ‘s’.
\aclp{ac} Works in the same way as \acl, but makes the long form into an English plural by adding an ‘s’.

### 2.2 Customization

The appearance of \acs and \acf can be configured in various ways. Of main importance are the package options:

- **footnote** makes the full name of the acronym appear as a footnote.
- **smaller** lets the acronyms appear a bit smaller than the surrounding text. This is in accord with typographic convention. The relsize package is required.

There are three lower-level macros controlling the output. Any acronym printed by \acs is formatted by \acsfont. Similarly, unless the option footnote is specified, \acffont handles the output of \acf, where the included acronym goes through \acfsfont (and \acsfont). The plural forms are treated accordingly. Usually the three macros do nothing. To give an example, the option smaller makes \acsfont use the command \textsmaller from the relsize package:

```
\renewcommand*{\acsfont}{\textsmaller{#1}}
```

### 2.3 Defining Acronyms

With the acronym environment you define all the acronyms used throughout your document. It is possible to add a longer description to each acronym definition.

In the acronym environment, acronyms are defined with the command:

```
\acro{(acronym)}[(short name)]{(full name)}{(explanation)}
```

The first argument (acronym) is the acronym string itself and is used in the commands of the previous section such as \ac or \acl, that print the different forms of the acronym.

Because internal commands take (acronym) for storing the different forms of the acronym, the \TeX code for the acronym is limited by \csname. If the acronym requires problematic or complicate \TeX stuff (font commands, \ldots), then this code can be given in the optional argument (short name). The first argument (acronym) is then a simpler string to identify the acronym. For example, an acronym for water can look like this:

```
\acro{H2O}[$\mathrm{H_2O}$]{water}
```

Then \acs{H2O} gets “H$_2$O” and \acl{H2O} prints “water”.

The acronym environment uses the description environment and the whole \acro definition acts like:

*This file has version v1.6 last revised 2000/05/21*
\item[(acronym)](full name) (explanation)

Or, if the optional argument (short name) is given, it looks like:

\item[(short name)](full name) (explanation)

\acrodef If you want to define acronyms which do not appear in the acronym environment, you can use the command:

\acrodef{(acronym)}{(short name)}{(full name)}

All acronym definitions, made by \acro or \acrodef are added to the .aux file. Therefore they are available from start-up in the next run. And the acronym definitions can be anywhere in the text.
3 An example file

\begin{document}
\section{Intro}
In the early nineties, \acs{GSM} was deployed in many European countries. \acs{GSM} offered for the first time international roaming for mobile subscribers. The \acs{GSM}'s use of \acs{TDMA} as its communication standard was debated at length. And every now and then there are big discussion whether \acs{CDMA} should have been chosen over \acs{TDMA}.

If you want to know more about \acs{GSM}, \acs{TDMA}, \acs{CDMA} and \oa, just read a book about mobile communication.

\subsection{Some chemistry and physics}
\acs{NAD+} is a major electron acceptor in the oxidation of fuel molecules. The reactive part of \acs{NAD+} is its nictinamide ring, a pyridine derivate.

One mol consists of \acs{NA} atoms or molecules. There is a relation between the constant of Boltzmann and the \ac{NA}:
\begin{equation}
k = R/\acs{NA}
\end{equation}

\section{Acronyms}
\begin{acronym}
\acro{GSM}{Global System for Mobile communication}.
\acro{GSM} is the new standard for digital cellular communication in Europe.
\acro{TDMA}{Time Division Multiple Access}.
Some would say, that this is not as good as \acs{CDMA}.
\acro{CDMA}{Code Division Multiple Access}. The spread spectrum modulation used in the Qualcomm system.
\acrodef{oa}{other acronyms}
\acro{NAD+}{Nicotinamide Adenine Dinucleotide}.
\acro{NA}{Number of Avogadro}:
\ensuremath{\acs{NA} = 6.022045\cdot10^{23}\,\text{mol}^{-1}}
\end{acronym}
\end{document}
4 The implementation

First we test that we got the right format and name the package.

\NeedsTeXFormat{LaTeX2e}[1994/06/01]
\ProvidesPackage{acronym}[iledate\space \fileversion\space Support for acronyms]

4.1 Defining acronyms

There are three commands that define acronyms: \texttt{\newacro}, \texttt{\acrodef}, and \texttt{\acro}.

They are called with the following arguments:

\acro{(acronym)}{(short name)}{(full name)}

There are two possibilities to scan the optional argument: It can be done by each command, or it can be done by a help command, that uses the information of the first argument to provide the optional argument, if it is not given. So the next command always have an optional argument. Here the second method is used, because it keeps the definition of the acronym defining commands simpler.

\newacro

The internal help macro \texttt{\AC@dblargafter} implements the second method:

\AC@dblargafter{first}\cmd[opt]... calls \cmd[first][opt]...
\AC@dblargafter{first}\cmd... gives \cmd[first][first]...

\acrodef

The user command \texttt{\acrodef} calls \texttt{\newacro} and writes it into the .aux file.

\newacro
I like to have a list of all acronyms I used in my document. Therefore you can define your acronyms inside the **acronym** environment. Not only stating the name of the acronym, but optionally also giving an explanation on it.

\begin{acronym}
\acro{CDMA}{Code Division Multiple Access}. The spread ...
\end{acronym}

**acro** Acronyms can be defined with the user command \acro{} in this **acronym** environment:

\begin{verbatim}
newenvironment{acronym}{% 
\begin{description} 
\providecommand*{\acro}[1]{\AC@dblargafter{##1}\AC@acro}% 
\}% \end{description} 
\}
newcommand\AC@acro{} 
def\AC@acro#1[#2]#3{% 
\acrodef{#1}[{#2}]{#3}% 
\item[{#2}] #3% 
}%
\end{verbatim}

**4.2 Using acronyms**

**\AC@check** If the acronym is undefined, the internal macro \AC@check{} warns the user and provides a default for the full name. The acronym is printed fat with an exclamation mark at the end.

\begin{verbatim}
newcommand+{\AC@check}[1]{% 
extendafter/\ifx/cname fn\#1\endcode\relax 
\PackageWarning{acronym}{Acronym ‘#1’ is not defined} 
\end{verbatim}

**\acsfont** The appearance of the output of the commands \acs{} and \acf{} is partially controlled by \acsfont, \acffont, and \acfsfont. By default, they do nothing.
\acs  The user macro \acs prints the short form of the acronym using the font specified by \acsfont. This is the acronym itself or the \textit{(short name)}, if the optional argument is given in the definition of the acronym. This \textit{(short name)} is available as second argument of the data in \texttt{fn\@<acronym>}. The help macro \AC@getsecond extracts this second argument out of a list of two or three arguments. The output goes through \acsfont.

\begin{verbatim}
102 \newcommand*{\acs}[1]{% 103 \AC@check{#1}% 104 \acsfont{% 105 \expandafter\expandafter\expandafter\AC@getsecond 106 \csname fn@#1\endcsname{#1}\@nil 107 }% 108 } 109 \newcommand\AC@getsecond{} 110 \long\def\AC@getsecond#1#2#3\@nil#2}
\end{verbatim}

\acl  The user macro \acl prints the full name of the acronym. It uses the \LaTeX macro \texttt{@car} to extract the first argument of the data, that are stored in the command \texttt{fn\@<acronym>},

\begin{verbatim}
111 \newcommand*{\acl}[1]{% 112 \AC@check{#1}% 113 \expandafter\expandafter\expandafter@car 114 \csname fn@#1\endcsname\@nil 115 }
\end{verbatim}

\acf  The user macro \acf always prints the full name with the acronym. The format depends on \texttt{acffont} and \texttt{acfsfont}, and on the option \texttt{footnote} handled below.

\begin{verbatim}
116 \newcommand*{\acf}[1]{% 117 \acffont{ 118 \acl{#1}% 119 \nolinebreak[3] % 120 \acfsfont{\acs{#1}}}% 121 }% 122 }
\end{verbatim}

\ac  The first time an acronym is accessed its Full Name (FN) is printed. The next time just (FN). This is done by \texttt{gdefining} the \texttt{ac@FN} to be \texttt{@empty} after its first use.

\begin{verbatim}
123 \newcommand{\ac}[1]{% 124 \expandafter\ifx\csname ac@#1\endcsname\relax % 125 \acf{#1}% 126 \global\expandafter\let\csname ac@#1\endcsname@empty 127 \else 128 \acs{#1}% 129 \fi 130 }
\end{verbatim}

\acsp  The user macro \acsp prints the plural short form of the acronym. This is the acronym itself or the \textit{(short name)}, if the optional argument is given in the definition of the acronym plus an `s’. This macro is a wrapper around \acs.

\begin{verbatim}
131 \newcommand*{\acsp}[1]{% 132 \acs{#1}s% 133 }
\end{verbatim}
\aclp The user macro \aclp prints the plural full name of the acronym. It is a wrapper around \acl.
\begin{verbatim}
134 \newcommand*{\aclp}[1]{
135 \acl{#1}s
136 \}
\end{verbatim}
\acfp The user macro \acfp always prints the plural full name with the plural of the acronym. The format depends on \acffont and \acfsfont, and on the option \footnote handled below.
\begin{verbatim}
137 \newcommand*{\acfp}[1]{
138 \acffont{
139 \aclp{#1}\
140 \nolinebreak[3] %
141 \acfsfont{\(\acs{#1}\))}\
142 \}
143 }
\end{verbatim}
\acp The first time an acronym is accessed Full Names (FNs) is printed. The next time just (FNs). This is done by \gdefining the \ac@FN to be \@empty after its first use.
\begin{verbatim}
144 \newcommand{\acp}[1]{
145 \expandafter\ifx\csname ac@#1\endcsname\relax
146 \acfp{#1}
147 \global\expandafter\let\csname ac@#1\endcsname\@empty
148 \else
149 \acs{#1}\
150 \fi
151 }
\end{verbatim}

4.3 Options

The option \footnote leads to a redefinition of \acf and \acfp, making the full name appear as a footnote. There is no need for \acffont and \acfsfont.
\begin{verbatim}
152 \DeclareOption{footnote}{% 
153 \renewcommand*{\acf}[1]{% 
154 \acs{#1}\
155 \footnote{\acl{#1}}%
156 }% 
157 \renewcommand*{\acfp}[1]{% 
158 \acsp{#1}\
159 \footnote{\aclp{#1}}%
160 }% 
161 }
\end{verbatim}
The option \smaller leads to a redefinition of \acsfont. We want to make the acronym appear smaller. Since this should be done in a context-sensitive way, we rely on the macro \textsmaller provided by the \textsize package. As \textPackage cannot be used inside \DeclareOption, we need a boolean variable.
\begin{verbatim}
162 \newif\if@ac@smaller
163 \@ac@smallerfalse
164 \DeclareOption{smaller}{\@ac@smallertrue}
\end{verbatim}
Now we process the options and care for the \texttt{smaller} option.

\begin{verbatim}
\ProcessOptions\relax
\if@ac@smaller
  \RequirePackage{relsize}
  \renewcommand*{\acsfont}{\textsmaller{#1}}
\fi

That's it.
\end{verbatim}

\endinput