

The HEP-ACRONYM package*

An acronym extension for glossaries

Jan Hajer†

2023/07/01

Abstract

The HEP-ACRONYM package provides an acronym macro based on the GLOSSARIES package.

Acronyms are implemented using the GLOSSARIES-EXTRA package [1] which is an extension of the GLOSSARIES package [2] and must be loaded after the HYPERREF package [3]. It can be loaded using `\usepackage{hep-acronym}`.

\acronym The `\acronym(*)[typeset abbreviation]{abbreviation}{*}{definition}{plural definition}` macro generates the singular `\langleabbreviation\rangle` and plural `\langleabbreviation\rangles` macros.

The first star prevents the addition of an ‘s’ to the abbreviation plural. The second star restores the TeX default of swallowing subsequent white space. The long form is only shown at the first appearance of these macros, later appearances generate the abbreviation with a hyperlink to the long form. Capitalisation at the beginning of paragraphs and sentences is (mostly) ensured. The long form is never used in math mode which can be exploited to enforce the short form. In order to enforce the long form use `\langleabbreviation\ranglelong`. As there can be no fixed rule whether to use the long form or the short form in section headers the user is left to their own devices e.g. `\glsdesc{abbreviation}`, `\Glsdesc{abbreviation}`. The **\sentence** macro ensures that the directly following abbreviation is capitalised.

\shortacronym The `\shortacronym` and `\longacronym` macros are drop-in replacements of the `\acronym` macro showing only the short or long form of their acronym.

\longacronym

\resetacronym The first use form of the acronym can be enforced by resetting the acronym counter with `\resetacronym{key}`.

\dummyacronym If the acronym counter equals one at the end of the document the short form of the acronym is not introduced. Placing a `\dummyacronym{key}` at the end of the document ensures that the short form is introduced.

warning In order to reduce the number of potentially conflicting packages the GLOSSARIES package is loaded without any glossary style. In the case that the glossary should be printed additional packages must be loaded via e.g. `\usepackage{glossary-list}`.

A Implementation

`<*package>`

Define a hepacronym namespace for the options using the KVOPTIONS package [4].

*This document corresponds to HEP-ACRONYM v1.2.

†jan.hajer@tecnico.ulisboa.pt

```

1 \RequirePackage{kvoptions}
2 \SetupKeyvalOptions{
3   family=hepacronym,
4   prefix=hepacronym@
5 }

```

hyper Define the hyper option controlling the hyperlink to the first appearance of the acronyms.

```
6 \DeclareBoolOption[true]{hyper}
```

Process options.

```
7 \ProcessKeyvalOptions*
```

Load the base package [1, 2] and set the abbreviation style.

```

8 \PassOptionsToPackage{nostyles}{glossaries-extra}
9 \RequirePackage{glossaries-extra}
10 % \setabbreviationstyle{long-hyphen-short-hyphen}

```

Use the entry count feature.

```

11 \glsenableentrycount
12 \glssetcategoryattribute{abbreviation}{entrycount}{1}

```

Provide macros for older glossaries-extra installations.

```

13 \AtBeginDocument{
14   \ifpackageloaded{hyperref}{
15     \providecommand{\glsxtrusefield}[2]{\@gls@entry@field{#1}{#2}}
16     \providecommand{\glsxtrsetfieldifexists}[3]{\glsdoifexists{#1}{#3}}
17     \providecommand{\gGlsXtrSetField}[3]{%
18       \glsxtrsetfieldifexists{#1}{#2}{%
19         \csgdef{glo@\glsdetoklabel{#1}@#2}{#3}%
20       }%
21   }

```

Set hyperlinks from the abbreviation to their definition in the text.

```

22   \glssetcategoryattribute{abbreviation}{nohyperfirst}{true}
23   \renewcommand*{\glsdonohyperlink}[2]{%
24     \glsxtrprotectlinks\edef\fieldvalue{%
25       \glsxtrusefield{\glslabel}{hastarget}%
26     }%
27     \ifdefstring\fieldvalue{true}{#2}{%
28       \gGlsXtrSetField{\glslabel}{hastarget}{true}%
29       \glsdohypertarget{#1}{#2}%
30     }%
31   }}
32 }{}
33 }

```

Provide for the case that hyperref is not loaded.

```
34 \newcommand{\hep@disable}[1]{%
35   \ifpackageloaded{hyperref}{\pdfstringdefDisableCommands{#1}}{}%
36 }
```

Deactivate hyperrefs if requested.

```
37 \ifhep@acronym@hyper\else
38   \glsdisablehyper
39 \fi
```

`\begin@sentence` Mark the beginning of a paragraph as if it would follow a full stop using the EVERYHOOK package [5].

```
40 \PassOptionsToPackage{excludeor}{everyhook}
41 \RequirePackage{everyhook}
42 \newcommand{\begin@sentence}{1001}
43 \newcommand{\sentence}{\spacefactor=\begin@sentence}
44 \hep@disable{\renewcommand\sentence{}}
45 \PushPostHook{par}{\sentence}
46 \gdef\ttl@case{\leavevmode\sentence}
```

`\frenchspacing` Adjust the `\frenchspacing` macro to be compatibel with this idea.

```
47 \def\frenchspacing{%
48   \sfcode'\.\begin@sentence \sfcode'\?\begin@sentence
49   \sfcode'\!\begin@sentence \sfcode'\:\begin@sentence
50   \sfcode'\;\@m \sfcode'\,\@m
51 }
```

`\if@begin@of@sentence` Provide a macro checking for the beginning of a sentence by examining the length of the preceding space.

```
52 \newcommand{\if@begin@of@sentence}[2]{\leavevmode\protecting{%
53   \ifboolexpr{ test {\ifnumcomp{\spacefactor}={}{3000}} or%
54               test {\ifnumcomp{\spacefactor}={}{2000}} or%
55               test {\ifnumcomp{\spacefactor}={}{\begin@sentence}}}%
56   }{#1}{#2}%
57 }}
```

`\mathdef` Provide a macro that allows to extend a macro definition into math mode using the XPARSE package [6].

```
58 \RequirePackage{xparse}
59 \ProvideDocumentCommand{\mathdef}{m0{0}m}{%
60   \expandafter\let\csname heptext\string#1\endcsname=#1
61   \expandafter\newcommand\csname hepmath\string#1\endcsname[#2]{#3}
62   \DeclareRobustCommand#1{%
63     \ifmmode
64       \expandafter\let\expandafter\next\csname%
65         hepmath\string#1\endcsname%
66     \else
```

```

67     \expandafter\let\expandafter\next\csname%
68     hep@text\string#1\endcsname%
69     \fi
70     \next
71 }%
72 }

```

`\acronym` Define the `\acronym(*)[\typeset abbreviation]{\abbreviation}{*}{\definition}{\plural definition}` macro using the `XSPACE` [7] and `AMSTEXT` [8] packages.

```

#1 star for omitting the 's' in the short plural
#2 optional typeset abbreviation
#3 mandatory abbreviation
#4 star for restoring the TeX default for space after text macros
#5 mandatory long form
#6 optional plural long form

```

```

73 \RequirePackage{xspace}
74 \RequirePackage{amstext}
75 \NewDocumentCommand{\acronym}{somsmo}{
76   \newabbreviation[
77     type=\acronymtype,
78     sort=#3,
79     \glshortpluralkey=\IfBooleanTF{#1}{#3}{\IfNoValueTF{#2}{#3s}{#2s}},
80     longplural=\IfNoValueTF{#6}{#5s}{#6}
81   ]{#3}{\IfNoValueTF{#2}{#3}{#2}}{#5}

```

Provide the singular acronym macro.

```

82 \expandafter\newcommand\csname#3\endcsname[1] []{%
83   \ifbegin@of@sentence{%
84     \ifglused{#3}{\cgl{s}{#3}[##1]}{\cGls{#3}[##1]}%
85     }{\cgl{s}{#3}[##1]}%
86     \ifnum\glentrycurrcount{#3}>1\relax
87       \IfBooleanTF{#4}{\@xspace}%
88       \else\@xspace\fi
89   }

```

Expand the singular acronym macro in portable document format (PDF) labels.

```

90 \hep@disable{\expandafter\def\csname#3\endcsname{%
91   \IfNoValueTF{#2}{#3}{#2} }%
92 }

```

Provide the singular acronym macro in math mode.

```

93 \expandafter\mathdef\csname#3\endcsname{%
94   \mathinner{\text{\glstrshort{#3}}}\@gl@s@increment@currcount{#3}%
95 }

```

Provide the plural acronym macro.

```

96 \expandafter\newcommand\csname#3s\endcsname[1] []{%

```

```

97   \if@begin@of@sentence{\cGlspl{#3}[##1]}\cglspl{#3}[##1]}%
98   \IfBooleanTF{#4}{\@xspace}%
99   }

```

Expand the plural acronym macro in PDF labels.

```

100  \hep@disable{\expandafter\def\csname#3s\endcsname{%
101    \IfBooleanTF{#1}{#3}{\IfNoValueTF{#2}{#3s}{#2s}} }%
102  }

```

Provide the plural acronym macro in math mode.

```

103  \expandafter\mathdef\csname#3s\endcsname{%
104    \mathinner{\text{\glsxtrshortpl{#3}}}\@gls@increment@currcount{#3}%
105  }

```

Provide an enforced long form.

```

106  \expandafter\newcommand\csname#3long\endcsname[1][1]{%
107    \if@begin@of@sentence{\Glsdesc*{#3}[##1]}\glsdesc*{#3}[##1]}%
108    \IfBooleanTF{#4}{\@xspace}%
109  }

```

Ensure that it works in PDFs.

```

110  \hep@disable{\expandafter\def\csname#3long\endcsname{#5 }}

```

Provide an enforced long plural form.

```

111  \expandafter\newcommand\csname#3slong\endcsname[1][1]{%
112    \if@begin@of@sentence{%
113      \Glsxtrlongpl*{#3}[##1]}\glsxtrlongpl*{#3}[##1]%
114    }%
115    \IfBooleanTF{#4}{\@xspace}%
116  }

```

Ensure that it works in PDFs.

```

117  \hep@disable{\expandafter\def\csname#3slong\endcsname{%
118    \IfNoValueTF{#6}{#5s}{#6} }%
119  }

```

End of `\acronym`.

```

120 }

```

`\shortacronym` The `\shortacronym` never expands into the long form.

```

121 \NewDocumentCommand{\shortacronym}{somsmo}{

```

Provide the singular acronym macro.

```

122  \expandafter\newcommand\csname#3\endcsname[1][1]{%
123    \IfNoValueTF{#2}{#3}{#2}\IfBooleanTF{#4}{\@xspace}##1%
124  }

```

Expand the singular acronym macro in PDF labels.

```
125 \hep@disable{\expandafter\def\csname#3\endcsname{%
126   \IfNoValueTF{#2}{#3}{#2} }%
127 }
```

Provide the singular acronym macro in math mode.

```
128 \expandafter\mathdef\csname#3\endcsname{%
129   \mathinner{\text{\IfNoValueTF{#2}{#3}{#2}}}%
130 }
```

Provide the plural acronym macro.

```
131 \expandafter\newcommand\csname#3s\endcsname[1][ ]{%
132   \IfBooleanTF{#1}{#3}{\IfNoValueTF{#2}{#3s}{#2s}}%
133   \IfBooleanTF{#4}{-}{\@xspace}##1%
134 }
```

Expand the plural acronym macro in PDF labels.

```
135 \hep@disable{\expandafter\def\csname#3s\endcsname{%
136   \IfBooleanTF{#1}{#3}{\IfNoValueTF{#2}{#3s}{#2s} }%
137 }
```

Provide the plural acronym macro in math mode.

```
138 \expandafter\mathdef\csname#3s\endcsname{%
139   \mathinner{\text{\IfBooleanTF{#1}{#3}{\IfNoValueTF{#2}{#3s}{#2s}}}}%
140 }
```

Ensure that long form also exists.

```
141 \expandafter\let\csname#3long\expandafter\endcsname\csname#3\endcsname
142 \expandafter\let\csname#3slong\expandafter\endcsname\csname#3s\endcsname
```

End of \shoracronym.

```
143 }
```

`\longacronym` The `\longacronym` never shows the abbreviated form.

```
144 \NewDocumentCommand{\longacronym}{somsmo}{
```

Provide the singular acronym macro.

```
145 \expandafter\newcommand\csname#3\endcsname[1][ ]{%
146   \if@begin@of@sentence{\MakeUppercase#5}{#5}%
147   \IfBooleanTF{#4}{-}{\@xspace}##1%
148 }
```

Expand the singular acronym macro in PDF labels.

```
149 \hep@disable{\expandafter\def\csname#3\endcsname{#5 } }
```

Provide the plural acronym macro.

```
150 \expandafter\newcommand\csname#3s\endcsname[1] [] {%
151   \if@begin@of@sentence{%
152     \IfNoValueTF{#6}{\MakeUppercase#5s}{\MakeUppercase#6}%
153   }{%
154     \IfNoValueTF{#6}{#5s}{#6}}\IfBooleanTF{#4}{}{\@xspace}##1%
155 }
```

Expand the plural acronym macro in PDF labels.

```
156 \hep@disable{\expandafter\def\csname#3s\endcsname{%
157   \IfNoValueTF{#6}{#5s}{#6} }%
158 }
```

Ensure that long form also exists.

```
159 \expandafter\let\csname#3long\expandafter\endcsname\csname#3\endcsname
160 \expandafter\let\csname#3slong\expandafter\endcsname\csname#3s\endcsname
```

End of `\longacronym`.

```
161 }
```

`\acronymalternative` Allow for alternative long text in abbreviation.

```
162 \NewDocumentCommand{\acronymalternative}{mmsm}{%
```

Store it in the `useri` field.

```
163   \GlsXtrSetField{#1}{useri}{#4}%
```

Define the acronym

```
164   \expandafter\newcommand\csname#2\endcsname[1] [] {%
```

Redirect the relevant macros to the `useri` field.

```
165   \let\hep@glssaccesslong\glssaccesslong%
166   \let\hep@Glsaccesslong\Glsaccesslong%
167   \let\glssaccesslong\glssuseri%
168   \let\Glsaccesslong\Glsuseri%
```

Execute abbreviation macros.

```
169   \if@begin@of@sentence{%
170     \ifglssused{#1}{\cglss{#1}[##1]}{\cGls{#1}[##1]}%
171     }{\cglss{#1}[##1]}%
```

Undo the redirection.

```
172   \let\glssaccesslong\hep@glssaccesslong%
173   \let\Glsaccesslong\hep@Glsaccesslong%
```

Spacing code.

```
174 \ifnum\glentrycurrcount{#1}>1\relax%
175 \IfBooleanTF{#3}{-}{\@xspace}%
176 \else\@xspace\fi%
177 }
```

Redirect PDF code to original acronym.

```
178 \hep@disable{\expandafter\def\csname#2\endcsname{%
179 \csname#1\endcsname }%
180 }
```

Redirect math macro to original acronym.

```
181 \expandafter\mathdef\csname#2\endcsname{%
182 \mathinner{\text{\glxtrshort{#1}}}\@gls@increment@currcount{#1}%
183 }
```

Provie long version of acronym.

```
184 \expandafter\newcommand\csname#2long\endcsname[1][ ]{%
```

Redirect relevant macros.

```
185 \let\hep@glentrydesc\glentrydesc%
186 \let\hep@Glsentrydesc\Glsentrydesc%
187 \renewrobustcmd*\glentrydesc[1]{\@gls@entry@field{#1}{useri}}%
188 \renewrobustcmd*\Glsentrydesc[1]{\@Gls@entry@field{#1}{useri}}%
```

Execute relevant macro.

```
189 \if@begin@of@sentence{\Glsdesc*{#1}[##1]}{\glsdesc*{#1}[##1]}%
190 \IfBooleanTF{#3}{-}{\@xspace}%
```

Undo redefinition.

```
191 \let\glentrydesc\hep@glentrydesc%
192 \let\Glsentrydesc\hep@Glsentrydesc%
193 }
```

Provie PDF code.

```
194 \hep@disable{%
195 \expandafter\def\csname#2long\endcsname{%
196 \@Gls@entry@field{#1}{useri} %
197 }%
198 }%
199 }
```

Silence warning if no acronyms are defined.

```
200 \renewcommand*\@gls@write@entrycounts{%
201 \immediate\write\@auxout{%
202 \string\providecommand*\string\@gls@entry@count}[2]{}
```



```

203 }%
204 \count@=0\relax
205 \forallglseentries{\@glseentry}{%
206   \glshasattribute{\@glseentry}{entrycount}{%
207     \ifglseused{\@glseentry}{%
208       \immediate\write\@auxout{%
209         \string\@glseentry@count{\@glseentry}{%
210           \glseentrycurrcount{\@glseentry}%
211         }
212       }%
213     }{} \advance\count@ by \@ne
214   }{}%
215 }%
216 }

```

`\resetacronym` Add two macros for acronym management.

`\dummyacronym`

```

217 \newcommand{\resetacronym}[1]{\protect\glsereset{#1}}
218 \newcommand{\dummyacronym}[1]{\protect\glseunset{#1}}

```

`abstract` Adjust the abstract environment to reset all acronym counters.

```

219 \@ifundefined{endabstract}{}{%
220   \let\end@hep@abstract\endabstract%
221   \renewcommand\endabstract{\glseresetall\end@hep@abstract}%
222 }

```

`\tableofcontents` Adjust the `\tableofcontents` macro to never show the long form of acronyms.

`\listoffigures`

`\listoftables`

```

223 \let\hep@table@of@contents\tableofcontents
224 \renewcommand\tableofcontents{%
225   \glseunsetall\hep@table@of@contents\glseresetall%
226 }
227 \let\hep@list@of@figures\listoffigures
228 \renewcommand\listoffigures{%
229   \glseunsetall\hep@list@of@figures\glseresetall%
230 }
231 \let\hep@list@of@tables\listoftables
232 \renewcommand\listoftables{%
233   \glseunsetall\hep@list@of@tables\glseresetall%
234 }

```

`\acronyms` Add a possibility to have different groups of acronyms.

```

235 \NewDocumentCommand{\acronyms}{om}{%
236   \IfNoValueTF{#1}{
237     \newglossary{#2}{#2.in}{#2.out}{#2}%
238     \renewcommand{\acronymtype}{#2}%
239   }{
240     \newglossary{#1}{#1.in}{#1.out}{#2}%
241     \renewcommand{\acronymtype}{#1}%

```

```
242 }
243 }

</package>
```

B Test

```
<*test>

244 \documentclass{article}
245
246 \usepackage{hyperref}
247 \usepackage{hep-acronym}
248 \usepackage{titlesec}
249
250 \acronym{PDF}{portable document format}
251 \acronym{URL}{uniform resource locator}
252 \acronym{CM}{computer modern}
253 \acronym{LM}{latin modern}
254
255 \begin{document}
256
257 \PDF \emph{(Capitalised at beginning of paragraph)}
258 \PDF \emph{(Second appearance is shortened)}.
259 \URLs \emph{(Capitalised plural form at beginning of sentence)}
260 $\URL$ \emph{(Second appearance is forced to be always short)}
261 \LM \emph{(Not capitalised in middle of sentence)}
262 \LM \emph{(Second appearance is shorted)}.
263 $\CM$ \emph{(Enforce short form)}
264 \CM \emph{(then extend second appearance)}
265 \URLlong \emph{(enforce long form)}
266
267 \section{\PDFlong table of contents is not broken}
268
269 \end{document}

</test>
```

C Readme

```
<*readme>

270 # The ‘hep-acronym’ package
271
272 An acronym extension for glossaries
273
274 ## Introduction
275
276 The ‘hep-acronym’ package provides an acronym macro based on the
277 ‘glossaries’ package. It can be loaded using ‘\usepackage{hep-acronym}’.
```

278
279 ## Author
280
281 Jan Hajer
282
283 ## License
284
285 This file may be distributed and/or modified under the conditions of the
286 'LaTeX' Project Public License, either version 1.3c of this license or
287 (at your option) any later version. The latest version of this license is
288 in '<http://www.latex-project.org/lppl.txt>' and version 1.3c or later is
289 part of all distributions of LaTeX version 2005/12/01 or later.

</readme>

References

- [1] N. T. Dickimaw Books. 'The `glossaries-extra` package: An extension to the `glossaries` package' (2015). CTAN: `glossaries-extra`.
- [2] N. Talbot. 'The `glossaries` package: Create glossaries and lists of acronyms' (2007). CTAN: `glossaries`.
- [3] *L^AT_EX₃ Project*. 'Hypertext marks in L^AT_EX: a manual for `hyperref`: Extensive support for hypertext in L^AT_EX' (1995). CTAN: `hyperref`. GitHub: `latex3/hyperref`.
- [4] H. Oberdiek. 'The `kvoptions` package: Key value format for package options' (2004). CTAN: `kvoptions`. GitHub: `ho-tex/kvoptions`.
- [5] S. Checkoway. 'The `everyhook` package: Hooks for standard T_EX token lists' (2010). CTAN: `everyhook`.
- [6] *L^AT_EX₃ Project*. 'The `xparse` package: A generic document command parser' (1999). CTAN: `xparse`.
- [7] *L^AT_EX Team*. 'The `xspace` package: Define commands that appear not to eat spaces' (1993). CTAN: `xspace`.
- [8] *L^AT_EX Team*. 'The `amstext` package: Typeset text fragments in mathematics' (1995). CTAN: `amstext`. URL: ams.org/tex/amslatex.