

The HEP-ACRONYM package*

An acronym extension for glossaries

Jan Hajer[†]

2026/08/01

Abstract

The HEP-ACRONYM package provides sentence-aware acronym helpers based on GLOSSARIES-EXTRA. It defines declaration commands for regular, always-short, always-long, and alternative long-form acronyms, optional links back to the first use, reset and dummy-use helpers, and support for several acronym groups.

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}`.

hyper The **hyper** option controls whether later acronym uses link back to their first appearance. It is enabled by default and can be deactivated with **hyper=false**.

\acronym The `\acronym{*{typeset abbreviation}{abbreviation}{*}{definition}{plural definition}}` macro generates the singular `\{abbreviation}` and plural `\{abbreviation}s` macros.

The first star prevents the addition of an ‘s’ to the abbreviation plural. The second star restores the T_EX 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 `\{abbreviation}long`. 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.

\sentence `\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

\acronymalternative The `\acronymalternative{acronym}{name}{*}{alternative long form}` macro defines an additional acronym macro that shares the use counter and short form of `acronym` but prints the supplied long form at first use.

\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.

\acronyms The `\acronyms{key}{title}` macro creates a new acronym glossary and switches subsequent

*This document corresponds to HEP-ACRONYM v1.5.

[†]jan.hajer@tecnico.ulisboa.pt

declarations to it. Without the optional argument the mandatory argument is used both as glossary key and printed title.

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].

```

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 \glsssetcategoryattribute{abbreviation}{entrycount}{1}

```

Provide macros for older `glossaries-extra` installations.

```

13 \AtBeginDocument{
14   \ifpackageloaded{hyperref}{
15     \providecommand*\glxtrusefield[2]{\@gls@entry@field{#1}{#2}}
16     \providecommand*\glxtrsetfieldifexists[3]{\glsdofexists{#1}{#3}}
17     \providecommand*\glsXtrSetField[3]{%
18       \glxtrsetfieldifexists{#1}{#2}{%
19         \csgdef{glo@glsdetoklabel{#1}#2}{#3}%
20       }%
21   }

```

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

```

22   \glsssetcategoryattribute{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 }
37 % Ignore only hyperref's "math shift" PDF-string warning.
38 % \begin{macrocode}
39 \AtBeginDocument{\%
40 \ifpackageloaded{hyperref}{\%
41 \let\hep@HyPsd@CatcodeWarning\HyPsd@CatcodeWarning
42 \renewcommand*{\HyPsd@CatcodeWarning}[1]{\%
43 \ifstrequal{#1}{math shift}{}{\}%
44 \hep@HyPsd@CatcodeWarning{#1}\%
45 }%
46 }%
47 }{\}%
48 }

```

Deactivate hyperrefs if requested.

```

49 \ifhep@acronym@hyper\else
50 \glsdisablehyper
51 \fi

```

Load the TITLESEC package [5] in order to ensure that section headers are treated as sentences.

```

52 \RequirePackage{titlesec}

```

`\begin@sentence` Mark the beginning of a paragraph as if it would follow a full stop using the EVERYHOOK package [6].

```

53 \PassOptionsToPackage{excludeor}{everyhook}
54 \RequirePackage{everyhook}
55 \newcommand*{\begin@sentence}{1001}
56 \newcommand*{\sentence}{\spacefactor=\begin@sentence}
57 \hep@disable{\renewcommand*\sentence{}}
58 \PushPostHook{par}{\sentence}
59 \gdef\ttl@case{\leavevmode\sentence}

```

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

```

60 \def\frenchspacing{%
61   \sfcode'\.\begin@sentence \sfcode'\?\begin@sentence
62   \sfcode'\!\begin@sentence \sfcode'\:\begin@sentence
63   \sfcode'\;\@m \sfcode'\,\@m
64 }

```

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

```

65 \newcommand*\hep@if@begin@of@sentence}[2]{\leavevmode\protecting{%
66   \ifboolexpr{ test {\ifnumcomp{\spacefactor}{=}{3000}} or%
67               test {\ifnumcomp{\spacefactor}{=}{2000}} or%
68               test {\ifnumcomp{\spacefactor}{=}{\begin@sentence}}}%
69   }{#1}{#2}%
70 }}

```

Fix the table of contents.

```

71 \let\hep@table@of@contents\tableofcontents
72 \renewcommand*\tableofcontents{%
73   \begingroup
74     \glsunsetall
75     \let\hep@contents@line\contentsline
76     \let\hep@number@line\numberline
77     \renewcommand*\contentsline[4]{%
78       \hep@contents@line{##1}{\sentence ##2}{##3}{##4}%
79     }%
80     \renewcommand*\numberline[1]{\hep@number@line{##1}\sentence}%
81     \hep@table@of@contents
82   \endgroup
83   \glsresetall
84 }

```

\hep@item@sentence Ensure that acronyms in description items get capitalised using the ENUMITEM package [7].
\descriptionsitem

```

85 \RequirePackage{enumitem}
86 \newcommand*\hep@item@sentence{\leavevmode\sentence}
87 \newcommand*\descriptionsitem{%
88   \let\hep@descripton@item\item
89   \RenewDocumentCommand{\item}{o}{%
90     \IfNoValueTF{##1}{%
91       \hep@descripton@item\hep@item@sentence
92     }{%
93       \hep@descripton@item[\hep@item@sentence##1]\hep@item@sentence
94     }%
95   }%
96 }
97 \setlist[description]{before=\descriptionsitem}

```

\mathdef Provide a macro that allows to extend a macro definition into math mode using the XPARSE package [8].

```

98 \RequirePackage{xparse}
99 \ProvideDocumentCommand{\mathdef}{mO{0}m}{%
100   \expandafter\let\csname hep@text\string#1\endcsname=#1
101   \expandafter\newcommand\expandafter*%
102     \csname hep@math\string#1\endcsname[#2]{#3}
103   \DeclareRobustCommand#1{%
104     \ifmmode
105       \expandafter\let\expandafter\next\csname%
106         hep@math\string#1\endcsname%
107     \else
108       \expandafter\let\expandafter\next\csname%
109         hep@text\string#1\endcsname%
110     \fi
111     \next
112   }%
113 }

```

`\acronym` Define the `\acronym(*)[typeset abbreviation]{abbreviation}{*}{definition}[plural definition]` macro using the `XSPACE` [9] and `AMSTEXT` [10] 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

```

114 \RequirePackage{xspace}
115 \RequirePackage{amstext}
116 \NewDocumentCommand{\acronym}{somsmo}{
117   \newabbreviation[
118     type=\acronymtype,
119     sort=#3,
120     \glsshortpluralkey=\IfBooleanTF{#1}{#3}{\IfNoValueTF{#2}{#3s}{#2s}},
121     longplural=\IfNoValueTF{#6}{#5s}{#6}
122   ]{#3}{\IfNoValueTF{#2}{#3}{#2}}{#5}

```

Provide the singular acronym macro.

```

123   \expandafter\newcommand\expandafter*\csname#3\endcsname[1][]{%
124     \hep@if@begin@of@sentence{%
125       \ifglssused{#3}{\cgl{s}{#3}[##1]}{\cgl{s}{#3}[##1]}%
126     }{\cgl{s}{#3}[##1]}%
127     \ifnum\glssentrycurrcount{#3}>1\relax
128       \IfBooleanTF{#4}{\@xspace}%
129     \else\@xspace\fi
130   }

```

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

```

131   \hep@disable{\expandafter\def\csname#3\endcsname{%
132     \IfNoValueTF{#2}{#3}{#2} }%
133 }

```

Provide the singular acronym macro in math mode.

```

134 \expandafter\mathdef\csname#3\endcsname{%
135   \mathinner{\text{\glstrshort{#3}}}\@gls@increment@currcount{#3}%
136 }

```

Provide the plural acronym macro.

```

137 \expandafter\newcommand\expandafter*\csname#3s\endcsname[1][1]{%
138   \hep@if@begin@of@sentence{\cGlspl{#3}[##1]}\cglsp1{#3}[##1]}%
139   \IfBooleanTF{#4}{-}{\@xspace}%
140 }

```

Expand the plural acronym macro in PDF labels.

```

141 \hep@disable{\expandafter\def\csname#3s\endcsname{%
142   \IfBooleanTF{#1}{#3}{\IfNoValueTF{#2}{#3s}{#2s}} }%
143 }

```

Provide the plural acronym macro in math mode.

```

144 \expandafter\mathdef\csname#3s\endcsname{%
145   \mathinner{\text{\glstrshortpl{#3}}}\@gls@increment@currcount{#3}%
146 }

```

Provide an enforced long form.

```

147 \expandafter\newcommand\expandafter*\csname#3long\endcsname[1][1]{%
148   \hep@if@begin@of@sentence{\Glsdesc*{#3}[##1]}\glsdesc*{#3}[##1]}%
149   \IfBooleanTF{#4}{-}{\@xspace}%
150 }

```

Ensure that it works in PDFs.

```

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

```

Provide an enforced long plural form.

```

152 \expandafter\newcommand\expandafter*\csname#3slong\endcsname[1][1]{%
153   \hep@if@begin@of@sentence{%
154     \Glsxtrlongpl*{#3}[##1]}\glxtrlongpl*{#3}[##1]%
155   }%
156   \IfBooleanTF{#4}{-}{\@xspace}%
157 }

```

Ensure that it works in PDFs.

```

158 \hep@disable{\expandafter\def\csname#3slong\endcsname{%
159   \IfNoValueTF{#6}{#5s}{#6} }%
160 }

```

Provide an enforced short form.

```

161 \expandafter\newcommand\expandafter*\csname#3short\endcsname[1][1]{%
162   \glstrshort{#3}\@gls@increment@currcount{#3}%

```

```

163     \IfBooleanTF{#4}{-}{\@xspace}%
164 }

```

Provide an enforced first form.

```

165 \expandafter\newcommand\expandafter*\csname#3first\endcsname[1][\{
166     \glsreset{#3}
167     \hep@if@begin@of@sentence{%
168         \ifglsused{#3}{\cgl{s}{#3}[##1]}{\cGls{#3}[##1]}%
169     }{\cgl{s}{#3}[##1]}%
170     \ifnum\glstentrycurrcount{#3}>1\relax
171         \IfBooleanTF{#4}{-}{\@xspace}%
172     \else\@xspace\fi
173 }

```

End of \acronym.

```

174 }

```

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

```

175 \NewDocumentCommand{\shortacronym}{somsmo}{

```

Provide the singular acronym macro.

```

176 \expandafter\newcommand\expandafter*\csname#3\endcsname[1][\{
177     \IfNoValueTF{#2}{#3}{#2}\IfBooleanTF{#4}{-}{\@xspace}##1%
178 }

```

Expand the singular acronym macro in PDF labels.

```

179 \hep@disable{\expandafter\def\csname#3\endcsname{%
180     \IfNoValueTF{#2}{#3}{#2} }%
181 }

```

Provide the singular acronym macro in math mode.

```

182 \expandafter\mathdef\csname#3\endcsname{%
183     \mathinner{\text{\IfNoValueTF{#2}{#3}{#2}}}%
184 }

```

Provide the plural acronym macro.

```

185 \expandafter\newcommand\expandafter*\csname#3s\endcsname[1][\{
186     \IfBooleanTF{#1}{#3}{\IfNoValueTF{#2}{#3s}{#2s}}%
187     \IfBooleanTF{#4}{-}{\@xspace}##1%
188 }

```

Expand the plural acronym macro in PDF labels.

```

189 \hep@disable{\expandafter\def\csname#3s\endcsname{%
190     \IfBooleanTF{#1}{#3}{\IfNoValueTF{#2}{#3s}{#2s}} }%
191 }

```

Provide the plural acronym macro in math mode.

```
192 \expandafter\mathdef\csname#3s\endcsname{%
193   \mathinner{\text{\IfBooleanTF{#1}{#3}{\IfNoValueTF{#2}{#3s}{#2s}}}}%
194 }%
```

Ensure that long form also exists.

```
195 \expandafter\let\csname#3long\expandafter\endcsname\csname#3\endcsname
196 \expandafter\let\csname#3slong\expandafter\endcsname\csname#3s\endcsname
```

End of \shoracronym.

```
197 }
```

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

```
198 \NewDocumentCommand{\longacronym}{\somsmo}{
```

Provide the singular acronym macro.

```
199 \expandafter\newcommand\expandafter*\csname#3\endcsname[1] [] {%
200   \hep@if@begin@of@sentence{\MakeUppercase#5}{#5}%
201   \IfBooleanTF{#4}{\@xspace}##1%
202 }
```

Expand the singular acronym macro in PDF labels.

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

Provide the plural acronym macro.

```
204 \expandafter\newcommand\expandafter*\csname#3s\endcsname[1] [] {%
205   \hep@if@begin@of@sentence{%
206     \IfNoValueTF{#6}{\MakeUppercase#5s}{\MakeUppercase#6}%
207   }{%
208     \IfNoValueTF{#6}{#5s}{#6}}\IfBooleanTF{#4}{\@xspace}##1%
209 }
```

Expand the plural acronym macro in PDF labels.

```
210 \hep@disable{\expandafter\def\csname#3s\endcsname{%
211   \IfNoValueTF{#6}{#5s}{#6} }%
212 }
```

Ensure that long form also exists.

```
213 \expandafter\let\csname#3long\expandafter\endcsname\csname#3\endcsname
214 \expandafter\let\csname#3slong\expandafter\endcsname\csname#3s\endcsname
```

End of \longacronym.

```
215 }
```

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

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

Store it in the `useri` field.

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

Define the acronym

```
218 \expandafter\newcommand\expandafter*\csname#2\endcsname[1][{}]{%
```

Redirect the relevant macros to the `useri` field.

```
219 \let\hep@glssaccesslong\glssaccesslong%
```

```
220 \let\hep@Glsaccesslong\Glsaccesslong%
```

```
221 \let\glssaccesslong\glssuseri%
```

```
222 \let\Glsaccesslong\Glsuseri%
```

Execute abbreviation macros.

```
223 \hep@if@begin@of@sentence{%
```

```
224 \ifglused{#1}{\cgl{#1}[##1]}{\cGls{#1}[##1]}%
```

```
225 }\cgl{#1}[##1]}%
```

Undo the redirection.

```
226 \let\glssaccesslong\hep@glssaccesslong%
```

```
227 \let\Glsaccesslong\hep@Glsaccesslong%
```

Spacing code.

```
228 \ifnum\glssentrycurrcount{#1}>1\relax%
```

```
229 \IfBooleanTF{#3}{\@xspace}%
```

```
230 \else\@xspace\fi%
```

```
231 }
```

Redirect PDF code to original acronym.

```
232 \hep@disable{\expandafter\def\csname#2\endcsname{%
```

```
233 \csname#1\endcsname }%
```

```
234 }
```

Redirect math macro to original acronym.

```
235 \expandafter\mathdef\csname#2\endcsname{%
```

```
236 \mathinner{\text{\glssxtrshort{#1}}}\@glss@increment@currcount{#1}%
```

```
237 }
```

Provide long version of acronym.

```
238 \expandafter\newcommand\expandafter*\csname#2long\endcsname[1][{}]{%
```

Redirect relevant macros.

```

239 \let\hep@glentrydesc\glentrydesc%
240 \let\hep@Glsentrydesc\Glsentrydesc%
241 \renewrobustcmd*{\glentrydesc}[1]{\@gls@entry@field{#1}{useri}}%
242 \renewrobustcmd*{\Glsentrydesc}[1]{\@Gls@entry@field{#1}{useri}}%

```

Execute relevant macro.

```

243 \hep@if@begin@of@sentence{\Glsdesc*{#1}[##1]}\@glsdesc*{#1}[##1]}%
244 \IfBooleanTF{#3}{-}{\@xspace}%

```

Undo redefinition.

```

245 \let\glentrydesc\hep@glentrydesc%
246 \let\Glsentrydesc\hep@Glsentrydesc%
247 }

```

Provie PDF code.

```

248 \hep@disable{%
249 \expandafter\def\csname#2long\endcsname{%
250 \@Gls@entry@field{#1}{useri} %
251 }%
252 }%
253 }

```

Silence warning if no acronyms are defined.

```

254 \renewcommand*{\@gls@write@entrycounts}{%
255 \immediate\write\@auxout{%
256 \string\providecommand*{\string\@gls@entry@count}[2]{%
257 }%
258 \count@=0\relax
259 \foralllglentries{\@glentry}{%
260 \glshasattribute{\@glentry}{entrycount}{%
261 \ifglused{\@glentry}{%
262 \immediate\write\@auxout{%
263 \string\@gls@entry@count{\@glentry}{%
264 \glentrycurrcount{\@glentry}%
265 }
266 }%
267 }{} \advance\count@ by \@ne
268 }{}%
269 }%
270 }

```

`\resetacronym` Add two macros for acronym management.

`\dummyacronym`

```

271 \newcommand*{\resetacronym}[1]{\protect\glsreset{#1}}
272 \newcommand*{\dummyacronym}[1]{\protect\glsunset{#1}}

```

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

```
273 \@ifundefined{endabstract}{\}%  
274 \let\end@hep@abstract\endabstract%  
275 \renewcommand*\endabstract{\glsresetall\end@hep@abstract}%  
276 }
```

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

```
\listoffigures  
\listoftables 277 % \let\hep@table@of@contents\tableofcontents  
278 % \renewcommand*\tableofcontents{%  
279 % \glsunsetall\hep@table@of@contents\glsresetall%  
280 % }  
281 % \let\hep@list@of@figures\listoffigures  
282 % \renewcommand*\listoffigures{%  
283 % \glsunsetall\hep@list@of@figures\glsresetall%  
284 % }  
285 % \let\hep@list@of@tables\listoftables  
286 % \renewcommand*\listoftables{%  
287 % \glsunsetall\hep@list@of@tables\glsresetall%  
288 % }
```

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

```
289 \NewDocumentCommand{\acronyms}{om}{%  
290 \IfNoValueTF{#1}{  
291 \newglossary{#2}{#2.in}{#2.out}{#2}%  
292 \renewcommand*\acronymtype{#2}%  
293 }{  
294 \newglossary{#1}{#1.in}{#1.out}{#2}%  
295 \renewcommand*\acronymtype{#1}%  
296 }  
297 }
```

</package>

B Test

<*test>

```
298 \documentclass{article}  
299  
300 \usepackage{hyperref}  
301 \usepackage{hep-acronym}  
302 \usepackage{titlesec}  
303  
304 \acronym{PDF}{portable document format}  
305 \acronym{URL}{uniform resource locator}  
306 \acronym{CM}{computer modern}  
307 \acronym{LM}{latin modern}  
308
```

```

309 \begin{document}
310
311 \PDF \emph{(Capitalised at beginning of paragraph)}
312 \PDF \emph{(Second appearance is shortened)}.
313 \URLs \emph{(Capitalised plural form at beginning of sentence)}
314 $\URL$ \emph{(Second appearance is forced to be always short)}
315 \LM \emph{(Not capitalised in middle of sentence)}
316 \LM \emph{(Second appearance is shorted)}.
317 $\CM$ \emph{(Enforce short form)}
318 \CM \emph{(then extend second appearance)}
319 \URLlong \emph{(enforce long form)}
320
321 \section{\PDFlong table of contents is not broken}
322
323 \end{document}

```

</test>

C Readme

```

<*readme>

324 # The 'hep-acronym' package
325
326 An acronym extension for glossaries
327
328 ## Introduction
329
330 The 'hep-acronym' package provides an acronym macro based on the
331 'glossaries-extra' package. It defines declaration commands for regular,
332 always-short, always-long, and alternative long-form acronyms, keeps the
333 first-use counters sentence-aware, optionally links later uses back to the
334 first occurrence, and provides helpers for resetting, dummy-using, and
335 grouping acronyms.
336
337 It can be loaded using '\usepackage{hep-acronym}'.
338
339 ## Author
340
341 Jan Hajer
342
343 ## License
344
345 This file may be distributed and/or modified under the conditions of the
346 'LaTeX' Project Public License, either version 1.3c of this license or
347 (at your option) any later version. The latest version of this license is
348 in 'http://www.latex-project.org/lppl.txt' and version 1.3c or later is
349 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] J. Bezos. ‘The `titlesec`, `titleps` and `titletoc` packages: Select alternative section titles’ (1998). CTAN: `titlesec`. GitHub: `jbezostitlesec`.
- [6] S. Checkoway. ‘The `everyhook` package: Hooks for standard T_EX token lists’ (2010). CTAN: `everyhook`.
- [7] J. Bezos. ‘Customizing lists with the `enumitem` package: Control layout of itemize, enumerate, description’ (2003). CTAN: `enumitem`. URL: texnia.com/enumitem.html. GitHub: `jbezosenumitem`.
- [8] *L^AT_EX₃ Project*. ‘The `xparse` package: A generic document command parser’ (1999). CTAN: `xparse`.
- [9] *L^AT_EX Team*. ‘The `xspace` package: Define commands that appear not to eat spaces’ (1993). CTAN: `xspace`.
- [10] *L^AT_EX Team*. ‘The `amstext` package: Typeset text fragments in mathematics’ (1995). CTAN: `amstext`. URL: ams.org/tex/amslatex.

Change History

v1.0	General: Created the dedicated acronym package with <code>glossaries-extra</code> based acronym declarations, sentence-aware use, short and long variants, reset and dummy helpers, and acronym groups. 1	README formatting. 1
v1.1	General: Added <code>\sentence</code> , forced long-form acronym commands, conditional PDF-string disabling without <code>hyperref</code> , safer option passing, and title-case tests. 1	v1.3 General: Renamed the sentence-state helper, applied starred command definitions, corrected release dates, and disabled the table/list acronym reset patch. 1
v1.2	General: Added the <code>hyper</code> option, alternative long-form acronyms, math-mode acronym atom wrapping, and refreshed documentation and	v1.4 General: Updated release metadata. 1 v1.5 General: Raised the LaTeX format requirement, bumped release metadata, and expanded the abstract, user documentation, README, and examples. 1