

The listingsutf8 package

Heiko Oberdiek*

2016/05/16 v1.3

Abstract

Package `listings` does not support files with multi-byte encodings such as UTF-8. In case of `\lstinputlisting` a simple workaround is possible if an one-byte encoding exists that the file can be converted to. Also ε -TeX and pdfTeX regardless of its mode are required.

Contents

1	Documentation	2
1.1	User interface	2
1.2	Future	2
2	Implementation	2
2.1	Catcodes and identification	2
2.2	Package options	3
2.3	Check prerequisites	4
2.4	Add support for UTF-8	4
2.4.1	Conversion	4
2.4.2	Convert CR/LF pairs to LF	5
2.4.3	Patch <code>\lst@InputListing</code>	5
3	Test	5
3.1	Catcode checks for loading	5
3.2	Test example for latin1	6
4	Installation	7
4.1	Download	7
4.2	Bundle installation	7
4.3	Package installation	7
4.4	Refresh file name databases	8
4.5	Some details for the interested	8
5	References	8
6	History	9
	[2007/10/22 v1.0]	9
	[2007/11/11 v1.1]	9
	[2011/11/10 v1.2]	9
	[2016/05/16 v1.3]	9

*Please report any issues at <https://github.com/ho-tex/oberdiek/issues>

1 Documentation

1.1 User interface

Load this package after or instead of package `listings` [2]. The package does not define own options and passes given options to package `listings`.

The syntax of package `listings`' key `inputencoding` is extended:

```
inputencoding=utf8/⟨one-byte-encoding⟩  
Example: inputencoding=utf8/latin1
```

That means the file is encoded in UTF-8 and can be converted to the given *⟨one-byte-encoding⟩*. The available encodings for *⟨one-byte-encoding⟩* are listed in section “1.2 Supported encodings” of package `stringenc`'s documentation [3]. Of course, the encoding must encode its characters with one byte exactly. This excludes the unicode encodings (`utf8`, `utf16`, ...).

Only `\lstinputlisting` is supported by the syntax extension of key `inputencoding`.

Internally package `listingsutf8` reads the file as binary file via primitives of pdfTeX (`\pdffiledump`). Then the file contents is converted as string using package `stringenc` and finally the string is read as virtual file by ε -TeX's `\scantokens`.

1.2 Future

Workarounds are not provided for

- `\lstinline`
- Environment `lstlisting`.
- Environments defined by `\lstnewenvironment`.

Perhaps someone will find time to extend package `listings` with full native support for UTF-8. Then this package would become obsolete.

2 Implementation

```
1 ⟨*package⟩
```

2.1 Catcodes and identification

```
2 \begingroup\catcode61\catcode48\catcode32=10\relax%  
3 \catcode13=5 % ^~M  
4 \endlinechar=13 %  
5 \catcode123=1 % {  
6 \catcode125=2 % }  
7 \catcode64=11 % @  
8 \def\x{\endgroup  
9 \expandafter\edef\csname lstU@AtEnd\endcsname{%  
10 \endlinechar=\the\endlinechar\relax  
11 \catcode13=\the\catcode13\relax  
12 \catcode32=\the\catcode32\relax  
13 \catcode35=\the\catcode35\relax  
14 \catcode61=\the\catcode61\relax  
15 \catcode64=\the\catcode64\relax  
16 \catcode123=\the\catcode123\relax  
17 \catcode125=\the\catcode125\relax  
18 }%
```

```

19 }%
20 \x\catcode61\catcode48\catcode32=10\relax%
21 \catcode13=5 % ^M
22 \endlinechar=13 %
23 \catcode35=6 % #
24 \catcode64=11 % @
25 \catcode123=1 % {
26 \catcode125=2 % }
27 \def\TMP@EnsureCode#1#2{%
28   \edef\lstU@AtEnd{%
29     \lstU@AtEnd
30     \catcode#1=\the\catcode#1\relax
31   }%
32   \catcode#1=#2\relax
33 }
34 \TMP@EnsureCode{10}{12}% ^^J
35 \TMP@EnsureCode{33}{12}% !
36 \TMP@EnsureCode{36}{3}% $
37 \TMP@EnsureCode{38}{4}% &
38 \TMP@EnsureCode{39}{12}% '
39 \TMP@EnsureCode{40}{12}% (
40 \TMP@EnsureCode{41}{12}% )
41 \TMP@EnsureCode{42}{12}% *
42 \TMP@EnsureCode{43}{12}% +
43 \TMP@EnsureCode{44}{12}% ,
44 \TMP@EnsureCode{45}{12}% -
45 \TMP@EnsureCode{46}{12}% .
46 \TMP@EnsureCode{47}{12}% /
47 \TMP@EnsureCode{58}{12}% :
48 \TMP@EnsureCode{60}{12}% <
49 \TMP@EnsureCode{62}{12}% >
50 \TMP@EnsureCode{91}{12}% [
51 \TMP@EnsureCode{93}{12}% ]
52 \TMP@EnsureCode{94}{7}% ^ (superscript)
53 \TMP@EnsureCode{95}{8}% _ (subscript)
54 \TMP@EnsureCode{96}{12}% `
55 \TMP@EnsureCode{124}{12}% |
56 \TMP@EnsureCode{126}{13}% ~ (active)
57 \edef\lstU@AtEnd{\lstU@AtEnd\noexpand\endinput}

```

Package identification.

```

58 \NeedsTeXFormat{LaTeX2e}
59 \ProvidesPackage{listingsutf8}%
60 [2016/05/16 v1.3 Allow UTF-8 in listings input (HO)]

```

2.2 Package options

Just pass options to package listings.

```

61 \DeclareOption*{%
62   \PassOptionsToPackage\CurrentOption{listings}%
63 }
64 \ProcessOptions*

```

Key inputencoding was introduced in version 2002/04/01 v1.0 of package listings.

```

65 \RequirePackage{listings}[2002/04/01]

```

Ensure that \inputencoding is provided.

```

66 \AtBeginDocument{%
67   \@ifundefined{inputencoding}{%
68     \RequirePackage{inputenc}%

```

```

69 }{}%
70 }

```

2.3 Check prerequisites

```

71 \RequirePackage{pdftexcmds}[2011/04/22]
72 \def\lstU@temp#1#2{%
73   \begingroup\expandafter\expandafter\expandafter\endgroup
74   \expandafter\ifx\csname #1\endcsname\relax
75     \PackageWarningNoLine{listingsutf8}{%
76       Package loading is aborted because of missing %
77       \@backslashchar#1.\MessageBreak
78       #2%
79     }%
80   \expandafter\lstU@AtEnd
81   \fi
82 }
83 \lstU@temp{scantokens}{It is provided by e-TeX}%
84 \lstU@temp{pdf@unescapehex}{It is provided by pdfTeX >= 1.30}%
85 \lstU@temp{pdf@filedump}{It is provided by pdfTeX >= 1.30}%
86 \lstU@temp{pdf@filesize}{It is provided by pdfTeX >= 1.30}%
87 \RequirePackage{stringenc}[2010/03/01]

```

2.4 Add support for UTF-8

```

\iflstU@utfviii
28 \newif\iflstU@utfviii

\lstU@inputenc
89 \def\lstU@inputenc#1{%
90   \expandafter\lstU@@inputenc#1utf8/utf8/\@nil
91 }

```

```

\lstU@@inputenc

92 \lst@Key{inputencoding}\relax{%
93   \def\lst@inputenc{#1}%
94   \lstU@inputenc{#1}%
95 }

```

2.4.1 Conversion

```

\lstU@input
96 \def\lstU@input#1{%
97   \iflstU@utfviii
98     \edef\lstU@text{%
99       \pdf@unescapehex{%
100         \pdf@filedump{0}{\pdf@filesize{#1}}{#1}%
101       }%
102     }%
103     \lstU@CRLFtoLF\lstU@text
104     \StringEncodingConvert\lstU@text\lstU@text{utf8}\lst@inputenc
105     \def\lstU@temp{%
106       \scantokens\expandafter{\lstU@text}%
107     }%
108   \else
109     \def\lstU@temp{%

```

```

110     \input{#1}%
111   }%
112   \fi
113   \lstU@temp
114 }

```

2.4.2 Convert CR/LF pairs to LF

\lstU@CRLFtoLF

```

115 \begingroup
116   \endlinechar=-1 %
117   \@makeother\^^J %
118   \@makeother\^^M %
119   \gdef\lstU@CRLFtoLF#1{%
120     \edef#1{%
121       \expandafter\lstU@CRLFtoLF@aux#1^^M^^J\@nil
122     }%
123   }%
124   \gdef\lstU@CRLFtoLF@aux#1^^M^^J#2\@nil{%
125     #1%
126     \ifx\relax#2\relax
127       \@car
128     \fi
129     ^^J%
130     \lstU@CRLFtoLF@aux#2\@nil
131   }%
132 \endgroup %

```

2.4.3 Patch \lst@InputListing

```

133 \def\lstU@temp#1\def\lst@next#2#3\@nil{%
134   \def\lst@InputListing##1{%
135     #1%
136     \def\lst@next{\lstU@input{##1}}%
137     #3%
138   }%
139 }
140 \expandafter\lstU@temp\lst@InputListing{#1}\@nil
141 \lstU@AtEnd%
142 \</package>

```

3 Test

3.1 Catcode checks for loading

```

143 \<{*test1}
144 \NeedsTeXFormat{LaTeX2e}
145 \documentclass{minimal}
146 \makeatletter
147 \def\RestoreCatcodes{}
148 \count@=0 %
149 \loop
150   \edef\RestoreCatcodes{%
151     \RestoreCatcodes
152     \catcode\the\count@=\the\catcode\count@\relax
153   }%
154 \ifnum\count@<255 %

```

```

155 \advance\count@\@ne
156 \repeat
157
158 \def\RangeCatcodeInvalid#1#2{%
159 \count@=#1\relax
160 \loop
161 \catcode\count@=15 %
162 \ifnum\count@<#2\relax
163 \advance\count@\@ne
164 \repeat
165 }
166 \def\Test{%
167 \RangeCatcodeInvalid{0}{47}%
168 \RangeCatcodeInvalid{58}{64}%
169 \RangeCatcodeInvalid{91}{96}%
170 \RangeCatcodeInvalid{123}{127}%
171 \catcode`\@=12 %
172 \catcode`\=0 %
173 \catcode`\{=1 %
174 \catcode`\}=2 %
175 \catcode`\#=6 %
176 \catcode`\[=12 %
177 \catcode`\]=12 %
178 \catcode`\%=14 %
179 \catcode`\ =10 %
180 \catcode13=5 %
181 \RequirePackage{listingsutf8}[2016/05/16]\relax
182 \RestoreCatcodes
183 }
184 \Test
185 \csname @@end\endcsname
186 \end
187 </test1>

```

3.2 Test example for latin1

```

188 (*test2)
189 \NeedsTeXFormat{LaTeX2e}
190 \documentclass{minimal}
191 \usepackage{filecontents}
192 \def\do#1{%
193 \ifx#1\~%
194 \else
195 \noexpand\do\noexpand#1%
196 \fi
197 }
198 \expandafter\let\expandafter\dospecials\expandafter\empty
199 \expandafter\edef\expandafter\dospecials\expandafter{\dospecials}
200 \begin{filecontents*}{ExampleUTF8.java}
201 public class ExampleUTF8 {
202     public static String testString =
203         "Umlauts: " +
204         "^^c3^^84^^c3^^96^^c3^^9c^^c3^^a4^^c3^^b6^^c3^^bc^^c3^^9f";
205     public static void main(String[] args) {
206         System.out.println(testString);
207     }
208 }
209 \end{filecontents*}

```

```

210 \usepackage{listingsutf8}[2016/05/16]
211 \def\Text{%
212   Umlauts: %
213   ^^c3^^84^^c3^^96^^c3^^9c^^c3^^a4^^c3^^b6^^c3^^bc^^c3^^9f%
214 }
215 \begin{document}
216 \lstinputlisting[%
217   language=Java,%
218   inputencoding=utf8/latin1,%
219 ]{ExampleUTF8.java}
220 \end{document}
221 \end{test2}

```

4 Installation

4.1 Download

Package. This package is available on CTAN¹:

[CTAN:macros/latex/contrib/oberdiek/listingsutf8.dtx](#) The source file.

[CTAN:macros/latex/contrib/oberdiek/listingsutf8.pdf](#) Documentation.

Bundle. All the packages of the bundle ‘oberdiek’ are also available in a TDS compliant ZIP archive. There the packages are already unpacked and the documentation files are generated. The files and directories obey the TDS standard.

[CTAN:install/macros/latex/contrib/oberdiek.tds.zip](#)

TDS refers to the standard “A Directory Structure for \TeX Files” ([CTAN:tds/tds.pdf](#)). Directories with `texmf` in their name are usually organized this way.

4.2 Bundle installation

Unpacking. Unpack the `oberdiek.tds.zip` in the TDS tree (also known as `texmf` tree) of your choice. Example (linux):

```
unzip oberdiek.tds.zip -d ~/texmf
```

Script installation. Check the directory `TDS:scripts/oberdiek/` for scripts that need further installation steps. Package `attachfile2` comes with the Perl script `pdfatfi.pl` that should be installed in such a way that it can be called as `pdfatfi`. Example (linux):

```

chmod +x scripts/oberdiek/pdfatfi.pl
cp scripts/oberdiek/pdfatfi.pl /usr/local/bin/

```

4.3 Package installation

Unpacking. The `.dtx` file is a self-extracting docstrip archive. The files are extracted by running the `.dtx` through plain \TeX :

```
tex listingsutf8.dtx
```

¹[CTAN:pkg/listingsutf8](#)

TDS. Now the different files must be moved into the different directories in your installation TDS tree (also known as `texmf` tree):

```
listingsutf8.sty      → tex/latex/oberdiek/listingsutf8.sty
listingsutf8.pdf      → doc/latex/oberdiek/listingsutf8.pdf
test/listingsutf8-test1.tex → doc/latex/oberdiek/test/listingsutf8-test1.tex
test/listingsutf8-test2.tex → doc/latex/oberdiek/test/listingsutf8-test2.tex
test/listingsutf8-test3.tex → doc/latex/oberdiek/test/listingsutf8-test3.tex
test/listingsutf8-test4.tex → doc/latex/oberdiek/test/listingsutf8-test4.tex
test/listingsutf8-test5.tex → doc/latex/oberdiek/test/listingsutf8-test5.tex
listingsutf8.dtx      → source/latex/oberdiek/listingsutf8.dtx
```

If you have a `docstrip.cfg` that configures and enables `docstrip`'s TDS installing feature, then some files can already be in the right place, see the documentation of `docstrip`.

4.4 Refresh file name databases

If your \TeX distribution (`te \TeX` , `mik \TeX` , ...) relies on file name databases, you must refresh these. For example, `te \TeX` users run `texhash` or `mktextlsr`.

4.5 Some details for the interested

Unpacking with \LaTeX . The `.dtx` chooses its action depending on the format:

plain \TeX : Run `docstrip` and extract the files.

\LaTeX : Generate the documentation.

If you insist on using \LaTeX for `docstrip` (really, `docstrip` does not need \LaTeX), then inform the autodetect routine about your intention:

```
latex \let\install=y\input{listingsutf8.dtx}
```

Do not forget to quote the argument according to the demands of your shell.

Generating the documentation. You can use both the `.dtx` or the `.drv` to generate the documentation. The process can be configured by the configuration file `ltxdoc.cfg`. For instance, put this line into this file, if you want to have A4 as paper format:

```
\PassOptionsToClass{a4paper}{article}
```

An example follows how to generate the documentation with `pdf \LaTeX` :

```
pdflatex listingsutf8.dtx
makeindex_sugind.ist listingsutf8.idx
pdflatex listingsutf8.dtx
makeindex_sugind.ist listingsutf8.idx
pdflatex listingsutf8.dtx
```

5 References

- [1] Alan Jeffrey, Frank Mittelbach, *inputenc.sty*, 2006/05/05 v1.1b. [CTAN:macros/latex/base/inputenc.dtx](#)
- [2] Carsten Heinz, Brooks Moses: *The listings package*; 2007/02/22; [CTAN:pkg/listings](#).
- [3] Heiko Oberdiek: *The stringenc package*; 2007/10/22; [CTAN:macros/latex/contrib/oberdiek/stringenc.pdf](#).

6 History

[2007/10/22 v1.0]

- First version.

[2007/11/11 v1.1]

- Use of package `pdftexcmds`.

[2011/11/10 v1.2]

- DOS line ends CR/LF normalized to LF to avoid empty lines (Bug report of Thomas Benkert in `de.comp.text.tex`).

[2016/05/16 v1.3]

- Documentation updates.

7 Index

Numbers written in *italic* refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; plain numbers refer to the code lines where the entry is used.

Symbols		
<code>\#</code>	175	<code>\count@</code> 148, 152, 154, 155, 159, 161, 162, 163
<code>\%</code>	178	<code>\csname</code> 9, 74, 185
<code>\@</code>	171	<code>\CurrentOption</code> 62
<code>\@backslashchar</code>	77	
<code>\@car</code>	127	D
<code>\@ifundefined</code>	67	<code>\DeclareOption</code> 61
<code>\@makeother</code>	117, 118	<code>\do</code> 192, 195
<code>\@ne</code>	155, 163	<code>\documentclass</code> 145, 190
<code>\@nil</code>	90, 121, 124, 130, 133, 140	<code>\dospecials</code> 198, 199
<code>\[</code>	176	E
<code>\]</code>	172	<code>\empty</code> 198
<code>\{</code>	173	<code>\end</code> 186, 209, 220
<code>\}</code>	174	<code>\endcsname</code> 9, 74, 185
<code>\]</code>	177	<code>\endinput</code> 57
<code>\^</code>	117, 118, 193	<code>\endlinechar</code> 4, 10, 22, 116
 <code>_</code>	179	G
		<code>\gdef</code> 119, 124
A		I
<code>\advance</code>	155, 163	<code>\iflstU@utfviii</code> 88, 97
<code>\AtBeginDocument</code>	66	<code>\ifnum</code> 154, 162
B		<code>\ifx</code> 74, 126, 193
<code>\begin</code>	200, 215	<code>\input</code> 110
C		L
<code>\catcode</code> 2, 3, 5, 6, 7, 11, 12, 13, 14, 15, 16, 17, 20, 21, 23, 24, 25, 26, 30, 32, 152, 161, 171, 172, 173, 174, 175, 176, 177, 178, 179, 180		<code>\loop</code> 149, 160
		<code>\lst@inputenc</code> 93, 104
		<code>\lst@InputListing</code> 134, 140
		<code>\lst@Key</code> 92

<code>\lst@next</code>	133, 136	<code>\ProvidesPackage</code>	59
<code>\lstinputlisting</code>	216		
<code>\lstU@@inputenc</code>	90, <u>92</u>		
<code>\lstU@AtEnd</code>	28, 29, 57, 80, 141		
<code>\lstU@CRLFtoLF</code>	103, <u>115</u>		
<code>\lstU@CRLFtoLF@aux</code>	121, 124, 130		
<code>\lstU@input</code>	<u>96</u> , 136		
<code>\lstU@inputenc</code>	<u>89</u> , 94		
<code>\lstU@temp</code>	72, 83,		
	84, 85, 86, 105, 109, 113, 133, 140		
<code>\lstU@text</code>	98, 103, 104, 106		
	M		
<code>\makeatletter</code>	146		
<code>\MessageBreak</code>	77		
	N		
<code>\NeedsTeXFormat</code>	58, 144, 189		
<code>\newif</code>	88		
	P		
<code>\PackageWarningNoLine</code>	75		
<code>\PassOptionsToPackage</code>	62		
<code>\pdf@filedump</code>	100		
<code>\pdf@filesize</code>	100		
<code>\pdf@unescapehex</code>	99		
<code>\ProcessOptions</code>	64		
		R	
		<code>\RangeCatcodeInvalid</code>	
			158, 167, 168, 169, 170
		<code>\repeat</code>	156, 164
		<code>\RequirePackage</code> ...	65, 68, 71, 87, 181
		<code>\RestoreCatcodes</code> ..	147, 150, 151, 182
		S	
		<code>\scantokens</code>	106
		<code>\StringEncodingConvert</code>	104
		T	
		<code>\Test</code>	166, 184
		<code>\Text</code>	211
		<code>\the</code> 10, 11, 12, 13, 14, 15, 16, 17, 30, 152	
		<code>\TMP@EnsureCode</code> .	27, 34, 35, 36, 37,
			38, 39, 40, 41, 42, 43, 44, 45, 46,
			47, 48, 49, 50, 51, 52, 53, 54, 55, 56
		U	
		<code>\usepackage</code>	191, 210
		X	
		<code>\x</code>	8, 20