The listingsutf8 package

Heiko Oberdiek
<oberdiek@uni-freiburg.de>

2007/10/22 v1.0

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

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

⟨*package⟩

2.1 Catcodes and identification

```
\expandafter\edef\csname lstU@AtEnd\endcsname{%
  \catcode64 \the\catcode64\relax
}
\catcode64 11 % @
\def\TMP@EnsureCode#1#2{%
  \edef\lstU@AtEnd{%
    \lstU@AtEnd
    \catcode#1 \the\catcode#1\relax
  }%
  \catcode#1 #2\relax
}
\TMP@EnsureCode{10}{12}% ^^J
\TMP@EnsureCode{33}{12}% !
\TMP@EnsureCode{36}{3}% $
\TMP@EnsureCode{38}{4}% &
\TMP@EnsureCode{39}{12}% '
\TMP@EnsureCode{40}{12}% (
\TMP@EnsureCode{41}{12}% )
\TMP@EnsureCode{42}{12}% *
\TMP@EnsureCode{43}{12}% +
\TMP@EnsureCode{44}{12}% ,
\TMP@EnsureCode{45}{12}% -
\TMP@EnsureCode{46}{12}% .
\TMP@EnsureCode{47}{12}% /
\TMP@EnsureCode{58}{12}% :
\TMP@EnsureCode{60}{12}% <
\TMP@EnsureCode{61}{12}% =
\TMP@EnsureCode{62}{12}% >
\TMP@EnsureCode{94}{7}% ^ (superscript)
```

```
\TMP@EnsureCode{95}{8}% _ (subscript)
\TMP@EnsureCode{96}{12}% ‘
\TMP@EnsureCode{124}{12}% |
\TMP@EnsureCode{126}{13}% ~ (active)
\g@addto@macro\lstU@AtEnd{\endinput}
```

Package identification.

```
\NeedsTeXFormat{LaTeX2e}
\ProvidesPackage{listingsutf8}%
  [2007/10/22 v1.0 Adding support for UTF-8 to listings (HO)]
```

2.2 Package options

Just pass options to package listings.

```
\DeclareOption*{%
  \PassOptionsToPackage\CurrentOption{listings}%
}
\ProcessOptions*
```

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

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

Ensure that `\inputencoding` is provided.

```
\AtBeginDocument{%
  \@ifundefined{inputencoding}{%
    \RequirePackage{inputenc}%
  }{}%
}
```

2.3 Check prerequisites

```
\def\lstU@temp#1#2{%
  \begingroup\expandafter\expandafter\expandafter\endgroup
  \expandafter\ifx\csname #1\endcsname\relax
    \PackageWarningNoLine{listingsutf8}{%
      Package loading is aborted because of missing %
      \@backslashchar#1.\MessageBreak
      #2%
    }%
    \expandafter\lstU@AtEnd
  \fi
}
\lstU@temp{scantokens}{It is provided by e-TeX}
\lstU@temp{pdfunescapehex}{It is provided by pdfTeX >= 1.30}
\lstU@temp{pdffiledump}{It is provided by pdfTeX >= 1.30}
\lstU@temp{pdffilesize}{It is provided by pdfTeX >= 1.30}
\RequirePackage{stringenc}[2007/10/22]
```

2.4 Add support for UTF-8

`\iflstU@utfviii`

```
\newif\iflstU@utfviii
```

`\lstU@inputenc`

```
\def\lstU@inputenc#1{%
  \expandafter\lstU@@inputenc#1utf8/utf8/\@nil
}
```

`\lstU@@inputenc`

```
\lst@Key{inputencoding}\relax{%
  \def\lst@inputenc{#1}%
  \lstU@inputenc{#1}%
}
```

2.4.1 Conversion

\lstU@input

```
\def\lstU@input#1{%
  \iflstU@utfviii
    \edef\lstU@text{%
      \pdfunescapehex{%
        \pdffiledump length\pdffilesize{#1}{#1}%
      }%
    }%
    \StringEncodingConvert\lstU@text\lstU@text{utf8}\lst@inputenc
    \def\lstU@temp{%
      \scantokens\expandafter{\lstU@text}%
    }%
  \else
    \def\lstU@temp{%
      \input{#1}%
    }%
  \fi
  \lstU@temp
}
```

2.4.2 Patch \lst@InputListing

```
\def\lstU@temp#1\def\lst@next#2#3\@nil{%
  \def\lst@InputListing##1{%
    #1%
    \def\lst@next{\lstU@input{##1}}%
    #3%
  }%
}
\expandafter\lstU@temp\lst@InputListing{#1}\@nil
\lstU@AtEnd
```

⟨/package⟩

3 Test

3.1 Catcode checks for loading

⟨*test1⟩

```
\NeedsTeXFormat{LaTeX2e}
\documentclass{minimal}
\makeatletter
\def\RestoreCatcodes{}
\count@=0 %
\loop
  \edef\RestoreCatcodes{%
    \RestoreCatcodes
    \catcode\the\count@=\the\catcode\count@\relax
  }%
\ifnum\count@<255 %
  \advance\count@\@ne
\repeat

\def\RangeCatcodeInvalid#1#2{%
  \count@=#1\relax
  \loop
    \catcode\count@=15 %
  \ifnum\count@<#2\relax
    \advance\count@\@ne
  \repeat
}
```

```
\def\Test{%
  \RangeCatcodeInvalid{0}{47}%
  \RangeCatcodeInvalid{58}{64}%
  \RangeCatcodeInvalid{91}{96}%
  \RangeCatcodeInvalid{123}{127}%
  \catcode`\@=12 %
  \catcode`\\=0 %
  \catcode`\{=1 %
  \catcode`\}=2 %
  \catcode`\#=6 %
  \catcode`\[=12 %
  \catcode`\]=12 %
  \catcode`\%=14 %
  \catcode`\ =10 %
  \catcode13=5 %
  \RequirePackage{listingsutf8}[2007/10/22]\relax
  \RestoreCatcodes
}
\Test
\csname @@end\endcsname
\end
```

⟨/test1⟩

3.2 Test example for latin1

⟨*test2⟩

```
\NeedsTeXFormat{LaTeX2e}
\documentclass{minimal}
\usepackage{filecontents}
\def\do#1{%
  \ifx#1\^%
  \else
    \noexpand\do\noexpand#1%
  \fi
}
\expandafter\let\expandafter\dospecials\expandafter\empty
\expandafter\edef\expandafter\dospecials\expandafter{\dospecials}
\begin{filecontents*}{ExampleUTF8.java}
public class ExampleUTF8 {
    public static String testString =
        "Umlauts: " +
        "^^c3^^84^^c3^^96^^c3^^9c^^c3^^a4^^c3^^b6^^c3^^bc^^c3^^9f";
    public static void main(String[] args) {
        System.out.println(testString);
    }
}
\end{filecontents*}
\usepackage{listingsutf8}[2007/10/22]
\def\Text{%
  Umlauts: %
  ^^c3^^84^^c3^^96^^c3^^9c^^c3^^a4^^c3^^b6^^c3^^bc^^c3^^9f%
}
\begin{document}
\lstinputlisting[%
  language=Java,%
  inputencoding=utf8/latin1,%
]{ExampleUTF8.java}
\end{document}
```

⟨/test2⟩

4 Installation

4.1 Download

Package. This package is available on CTAN[1]:

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:macros/latex/contrib/oberdiek/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
```

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 (teTEX, mikTEX, ...) relies on file name databases, you must refresh these. For example, teTEX users run `texhash` or `mktexlsr`.

[1] ftp://ftp.ctan.org/tex-archive/

4.5 Some details for the interested

Attached source. The PDF documentation on CTAN also includes the `.dtx` source file. It can be extracted by AcrobatReader 6 or higher. Another option is pdftk, e.g. unpack the file into the current directory:

```
pdftk listingsutf8.pdf unpack_files output .
```

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 pdfLaTeX:

```
pdflatex listingsutf8.dtx
makeindex -s gind.ist listingsutf8.idx
pdflatex listingsutf8.dtx
makeindex -s gind.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:macros/latex/contrib/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.

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; numbers in roman refer to the code lines where the entry is used.