The selinput package

Heiko Oberdiek*
<heiko.oberdiek at googlemail.com>

2016/05/17 v1.4

Abstract

This package selects the input encoding by specifying between input characters and their glyph names.

Contents

1 Documentation

1.1 Introduction

LaTeX supports the direct use of 8-bit characters by means of package inputenc. However you must know and specify the encoding, e.g.:

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

```
\documentclass{article}
\usepackage[latin1]{inputenc}
% or \usepackage[utf8]{inputenc}
% or \usepackage[??]{inputenc}
\begin{document}
  Umlauts: ÄÖÜäöüß
\end{document}
```

If the document is transferred in an environment that uses a different encoding, then there are programs that convert the input characters. Examples for conversion of file `test.tex` from encoding latin1 (ISO-8859-1) to UTF-8:

```
recode ISO-8859-1..UTF-8 test.tex
recode latin1..utf8 test.tex
iconv --from-code ISO-8859-1
      --to-code UTF-8
      --output testnew.tex
      test.tex
iconv -f latin1 -t utf8 -o testnew.tex test.tex
```

However, the encoding name for package inputenc must be changed:

`\usepackage[latin1]{inputenc}` → `\usepackage[utf8]{inputenc}`

Of course, unless you are using some clever editor that knows package inputenc, recodes the file and adjusts the option at the same time. But most editors can perhaps recode the file, but they let the option untouched.

Therefore package selinput chooses another way for specifying the input encoding. The encoding name is not needed at all. Some 8-bit characters are identified by their glyph name and the package chooses an appropriate encoding, example:

```
\documentclass{article}
\usepackage{selinput}
\SelectInputMappings{
  adieresis={ä},
  germandbls={ß},
  Euro={€},
}
\begin{document}
  Umlauts: ÄÖÜäöüß
\end{document}
```

1.2 User interface

`\SelectInputEncodingList {`⟨*encoding list*⟩`}`

`\SelectInputEncodingList` expects a comma separated list of encoding names. Example:

```
\SelectInputEncodingList{utf8,ansinew,mac-roman}
```

The encodings of package inputenx are used as default.

`\SelectInputMappings {`⟨*mapping pairs*⟩`}`

A mapping pair consists of a glyph name and its input character:

```
\SelectInputMappings{
  adieresis={ä},
  germandbls={ß},
  Euro={€},
}
```

The supported glyph names can be found in file `ix-name.def` of project inputenx [1]. The names are basically taken from Adobe's glyphlists [2, 3]. As many pairs are needed as necessary to identify the encoding. Example with insufficient pairs:

```
\SelectInputEncodingSet{latin1,latin9}
\SelectInputMappings{
  adieresis={ä},
  germandbls={ß},
}
Umlauts: ÄÖÜäöüß and Euro: ¤ (wrong)
```

The first encoding latin1 passes the constraints given by the mapping pairs. However the Euro symbol is not part of the encoding. Thus a mapping pair with the Euro symbol solves the problem. In fact the symbol alone already succeeds in selecting between latin1 and latin9:

```
\SelectInputEncodingSet{latin1,latin9}
\SelectInputMappings{
  Euro={€},
}
Umlauts: ÄÖÜäöüß and Euro: €
```

1.3 Options

warning: The selected encoding is written by `\PackageInfo` into the `.log` file only. Option warning changes it to `\PackageWarning`. Then the selected encoding is shown on the terminal as well.

ucs: The encoding file `utf8x` of package `\ucs` requires that the package itself is loaded before. If the package is not loaded, then the option ucs will load package ucs if the detected encoding is UTF-8 (limited to the preamble, packages cannot be loaded later).

utf8=...: The option allows to specify other encoding files for UTF-8 than LaTeX's `utf8.def`. For example, `utf8=utf-8` will load `utf-8.def` instead.

1.4 Encodings

Package stringenc [4] is used for testing the encoding. Thus the encoding name must be known by this package. Then the found encoding is loaded by `\inputencoding` by package inputenc or `\InputEncoding` if package inputenx is loaded.

The supported encodings are present in the encoding list, thus usually the encoding names do not matter. If the list is set by `\SelectInputEncodingList`, then you can use the names that work for package inputenc and are known by package stringenc, for example: latin1, x-iso-8859-1. Encoding file names of package inputenx are prefixed with `x-`. The prefix can be dropped, if package inputenx is loaded.

2 Implementation

```
1 ⟨*package⟩
2 \NeedsTeXFormat{LaTeX2e}
3 \ProvidesPackage{selinput}
4   [2016/05/17 v1.4 Semi-automatic input encoding detection (HO)]%
5 \RequirePackage{inputenc}
6 \RequirePackage{kvsetkeys}[2006/10/19]
7 \RequirePackage{stringenc}[2007/06/16]
8 \RequirePackage{kvoptions}
```

\SelectInputEncodingList

```
\newcommand*{\SelectInputEncodingList}{%
  \let\SIE@EncodingList\@empty
  \kvsetkeys{SelInputEnc}%
}
```

\SelectInputMappings

```
\newcommand*{\SelectInputMappings}[1]{%
  \SIE@LoadNameDefs
  \let\SIE@StringUnicode\@empty
  \let\SIE@StringDest\@empty
  \kvsetkeys{SelInputMap}{#1}%
  \ifx\\\SIE@StringUnicode\SIE@StringDest\\%
    \PackageError{selinput}{%
      No mappings specified%
    }\@ehc
  \else
    \EdefUnescapeHex\SIE@StringUnicode\SIE@StringUnicode
    \let\SIE@Encoding\@empty
    \@for\SIE@EncodingTest:=\SIE@EncodingList\do{%
      \ifx\SIE@Encoding\@empty
        \StringEncodingConvertTest\SIE@temp\SIE@StringUnicode
                                  {utf16be}\SIE@EncodingTest{%
          \ifx\SIE@temp\SIE@StringDest
            \let\SIE@Encoding\SIE@EncodingTest
          \fi
        }{}%
      \fi
    }%
    \ifx\SIE@Encoding\@empty
      \StringEncodingConvertTest\SIE@temp\SIE@StringDest
                                {ascii}{utf16be}{%
        \def\SIE@Encoding{ascii}%
        \SIE@Info{selinput}{%
          Matching encoding not found, but input characters%
          \MessageBreak
          are 7-bit (possibly editor replacements).%
          \MessageBreak
          Hence using ascii encoding%
        }%
      }{}%
    \fi
    \ifx\SIE@Encoding\@empty
      \PackageError{selinput}{%
        Cannot find a matching encoding%
      }\@ehd
    \else
      \ifx\SIE@Encoding\SIE@EncodingUTFviii
        \SIE@LoadUnicodePackage
        \ifx\SIE@UseUTFviii\@empty
        \else
          \let\SIE@Encoding\SIE@UseUTFviii
        \fi
      \fi
      \begingroup\expandafter\expandafter\expandafter\endgroup
      \expandafter\ifx\csname InputEncoding\endcsname\relax
        \inputencoding\SIE@Encoding
      \else
        \InputEncoding\SIE@Encoding
      \fi
      \SIE@Info{selinput}{Encoding `\SIE@Encoding' selected}%
    \fi
```

```
  \fi
}
```

\SIE@LoadNameDefs

```
\def\SIE@LoadNameDefs{%
  \begingroup
    \endlinechar=\m@ne
    \catcode92=0 % backslash
    \catcode123=1 % left curly brace/beginning of group
    \catcode125=2 % right curly brace/end of group
    \catcode37=14 % percent/comment character
    \@makeother\[%
    \@makeother\]%
    \@makeother\.%
    \@makeother\(%
    \@makeother\)%
    \@makeother\/%
    \@makeother\-%
    \let\InputenxName\SelectInputDefineMapping
    \InputIfFileExists{ix-name.def}{}{%
      \PackageError{selinput}{%
        Missing `ix-name.def' (part of package `inputenx')%
      }\@ehd
    }%
    \global\let\SIE@LoadNameDefs\relax
  \endgroup
}
```

\SelectInputDefineMapping

```
\newcommand*{\SelectInputDefineMapping}[1]{%
  \expandafter\gdef\csname SIE@@#1\endcsname
}

\kv@set@family@handler{SelInputMap}{%
  \@onelevel@sanitize\kv@key
  \ifx\kv@value\relax
    \PackageError{selinput}{%
      Missing input character for `\kv@key'%
    }\@ehc
  \else
    \@onelevel@sanitize\kv@value
    \ifx\kv@value\@empty
      \PackageError{selinput}{%
        Input character got lost?\MessageBreak
        Missing input character for `\kv@key'%
      }\@ehc
    \else
      \@ifundefined{SIE@@\kv@key}{%
        \PackageWarning{selinput}{%
          Missing definition for `\kv@key'%
        }%
      }{%
        \edef\SIE@StringDest{%
          \SIE@StringDest
          \kv@value
        }%
        \edef\SIE@StringUnicode{%
          \SIE@StringUnicode
          \csname SIE@@\kv@key\endcsname
        }%
      }%
    \fi
  \fi
```

```
}
\kv@set@family@handler{SelInputEnc}{%
  \@onelevel@sanitize\kv@key
  \ifx\kv@value\relax
    \ifx\SIE@EncodingList\@empty
      \let\SIE@EncodingList\kv@key
    \else
      \edef\SIE@EncodingList{\SIE@EncodingList,\kv@key}%
    \fi
  \else
    \@onelevel@sanitize\kv@value
    \PackageError{selinput}{%
      Illegal key value pair (\kv@key=\kv@value)\MessagBreak
      in encoding list%
    }\@ehc
  \fi
}
```

\SIE@LoadUnicodePackage

```
\def\SIE@LoadUnicodePackage{%
  \@ifpackageloaded\SIE@UnicodePackage{}{%
    \RequirePackage\SIE@UnicodePackage\relax
  }%
  \SIE@PatchUCS
  \global\let\SIE@LoadUnicodePackage\relax
}
\let\SIE@show\show
\def\SIE@PatchUCS{%
  \AtBeginDocument{%
    \expandafter\ifx\csname ver@ucsencs.def\endcsname\relax
    \else
      \let\show\SIE@show
    \fi
  }%
}
\SIE@PatchUCS

\AtBeginDocument{%
  \let\SIE@LoadUnicodePackage\relax
}
```

\SIE@EncodingUTFviii

```
\def\SIE@EncodingUTFviii{utf8}
\@onelevel@sanitize\SIE@EncodingUTFviii
```

\SIE@EncodingUTFviiix

```
\def\SIE@EncodingUTFviiix{utf8x}
\@onelevel@sanitize\SIE@EncodingUTFviiix

\let\SIE@UnicodePackage\@empty
\let\SIE@UseUTFviii\@empty
\let\SIE@Info\PackageInfo

\SetupKeyvalOptions{%
  family=SelInput,%
  prefix=SelInput@%
}
\define@key{SelInput}{utf8}{%
  \def\SIE@UseUTFviii{#1}%
  \@onelevel@sanitize\SIE@UseUTFviii
}
\DeclareBoolOption{ucs}
\DeclareVoidOption{warning}{%
```

```
  \let\SIE@Info\PackageWarning
}
\ProcessKeyvalOptions{SelInput}
\ifSelInput@ucs
  \def\SIE@UnicodePackage{ucs}%
  \ifx\SIE@UseUTFviii\@empty
    \let\SIE@UseUTFviii\SIE@EncodingUTFviiix
  \fi
\else
  \ifx\SIE@UseUTFviii\@empty
    \@ifpackageloaded{ucs}{%
      \let\SIE@UseUTFviii\SIE@EncodingUTFviiix
    }{%
      \let\SIE@UseUTFviii\SIE@EncodingUTFviii
    }%
  \fi
\fi
```

\SIE@EncodingList

```
\edef\SIE@EncodingList{%
  utf8,%
  x-iso-8859-1,%
  x-iso-8859-15,%
  x-cp1252,% ansinew
  x-mac-roman,%
  x-iso-8859-2,%
  x-iso-8859-3,%
  x-iso-8859-4,%
  x-iso-8859-5,%
  x-iso-8859-6,%
  x-iso-8859-7,%
  x-iso-8859-8,%
  x-iso-8859-9,%
  x-iso-8859-10,%
  x-iso-8859-11,%
  x-iso-8859-13,%
  x-iso-8859-14,%
  x-iso-8859-15,%
  x-mac-centeuro,%
  x-mac-cyrillic,%
  x-koi8-r,%
  x-cp1250,%
  x-cp1251,%
  x-cp1257,%
  x-cp437,%
  x-cp850,%
  x-cp852,%
  x-cp855,%
  x-cp858,%
  x-cp865,%
  x-cp866,%
  x-nextstep,%
  x-dec-mcs%
}%
\@onelevel@sanitize\SIE@EncodingList
```

⟨/package⟩

3 Test

```
⟨*test⟩
\NeedsTeXFormat{LaTeX2e}
```

```
\documentclass{minimal}
\usepackage{textcomp}
\usepackage{qstest}
⟨*test1 j test2 j test3⟩
\makeatletter
\let\BeginDocumentText\@empty
\def\TestEncoding#1#2{%
  \SelectInputMappings{#2}%
  \Expect*{\SIE@Encoding}{#1}%
  \Expect*{\inputencodingname}{#1}%
  \g@addto@macro\BeginDocumentText{%
    \SelectInputMappings{#2}%
    \Expect*{\SIE@Encoding}{#1}%
    \textbf{\SIE@Encoding:} %
    \kvsetkeys{test}{#2}\par
  }%
}
\def\TestKey#1#2{%
  \define@key{test}{#1}{%
    \sbox0{##1}%
    \sbox2{#2}%
    \Expect*{wd:\the\wd0, ht:\the\ht0, dp:\the\dp0}%
            *{wd:\the\wd2, ht:\the\ht2, dp:\the\dp2}%
    [#1=##1] % hash-ok
  }%
}
\RequirePackage{keyval}
\TestKey{adieresis}{\"a}
\TestKey{germandbls}{\ss}
\TestKey{Euro}{\texteuro}
\makeatother
\usepackage[
  warning,%
⟨test2⟩  utf8=utf-8,
⟨test3⟩  ucs,
]{selinput}
⟨test1 j test3⟩\inputencoding{ascii}
⟨test2⟩\inputencoding{utf-8}
⟨test3⟩\usepackage{ucs}
\begin{qstest}{preamble}{}
  \TestEncoding{x-iso-8859-15}{%
    adieresis=^^e4,%
    germandbls=^^df,%
    Euro=^^a4,%
  }%
  \TestEncoding{x-cp1252}{%
    adieresis=^^e4,%
    germandbls=^^df,%
    Euro=^^80,%
  }%
⟨test1⟩  \TestEncoding{utf8}{%
⟨test2⟩  \TestEncoding{utf-8}{%
⟨test3⟩  \TestEncoding{utf8x}{%
    adieresis=^^c3^^a4,%
    germandbls=^^c3^^9f,%
⟨!test2⟩    Euro=^^e2^^82^^ac,
  }%
\end{qstest}
⟨test3⟩\let\ifUnicodeOptiongraphics\iffalse
\begin{document}
\begin{qstest}{document}{}
⟨test3⟩\makeatletter
```

```
  \BeginDocumentText
\end{qstest}
⟨/test1 | test2 | test3⟩
⟨*test4⟩
\usepackage[warning,ucs]{selinput}
\SelectInputMappings{%
    adieresis=^^c3^^a4,%
    germandbls=^^c3^^9f,%
    Euro=^^e2^^82^^ac,%
}
\begin{qstest}{encoding}{}
  \Expect*{\inputencodingname}{utf8x}%
\end{qstest}
\begin{document}
  adieresis=^^c3^^a4, %
  germandbls=^^c3^^9f, %
  Euro=^^e2^^82^^ac%
⟨/test4⟩
⟨*test5⟩
\usepackage[warning,ucs]{selinput}
\SelectInputMappings{%
    adieresis={\"a},%
    germandbls={{\ss}},%
    Euro=\texteuro{},%
}
\begin{qstest}{encoding}{}
  \Expect*{\inputencodingname}{ascii}%
\end{qstest}
\begin{document}
  adieresis={\"a}, %
  germandbls={{\ss}}, %
  Euro=\texteuro{}%
⟨/test5⟩
\end{document}
⟨/test⟩
```

4 Installation

4.1 Download

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

`CTAN:macros/latex/contrib/oberdiek/selinput.dtx` The source file.

`CTAN:macros/latex/contrib/oberdiek/selinput.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
```

[1] http://ctan.org/pkg/selinput

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 selinput.dtx
```

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

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

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{selinput.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 selinput.dtx
makeindex -s gind.ist selinput.idx
pdflatex selinput.dtx
makeindex -s gind.ist selinput.idx
pdflatex selinput.dtx
```

5 Catalogue

The following XML file can be used as source for the TEX Catalogue. The elements `caption` and `description` are imported from the original XML file from the Catalogue. The name of the XML file in the Catalogue is `selinput.xml`.

```
⟨*catalogue⟩
<?xml version='1.0' encoding='us-ascii'?>
<!DOCTYPE entry SYSTEM 'catalogue.dtd'>
<entry datestamp='$Date$' modifier='$Author$' id='selinput'>
  <name>selinput</name>
  <caption>Semi-automatic detection of input encoding.</caption>
  <authorref id='auth:oberdiek'/>
  <copyright owner='Heiko Oberdiek' year='2007'/>
  <license type='lppl1.3'/>
  <version number='1.4'/>
  <description>
    This package selects the input encoding by specifying pairs
    of input characters and their glyph names.
    <p/>
    The package is part of the <xref refid='oberdiek'>oberdiek</xref>
    bundle.
  </description>
  <documentation details='Package documentation'
      href='ctan:/macros/latex/contrib/oberdiek/selinput.pdf'/>
  <ctan file='true' path='/macros/latex/contrib/oberdiek/selinput.dtx'/>
  <miktex location='oberdiek'/>
  <texlive location='oberdiek'/>
  <install path='/macros/latex/contrib/oberdiek/oberdiek.tds.zip'/>
</entry>
⟨/catalogue⟩
```

6 References

[1] Heiko Oberdiek: *The inputenx package*; 2007-04-11 v1.1; CTAN:macros/latex/contrib/oberdiek/inputenx.pdf.

[2] Adobe: *Adobe Glyph List*; 2002-09-20 v2.0; http://partners.adobe.com/public/developer/en/opentype/glyphlist.txt.

[3] Adobe: *Adobe Glyph List For New Fonts*; 2005-11-18 v1.5; http://partners.adobe.com/public/developer/en/opentype/aglfn13.txt.

[4] Heiko Oberdiek: *The stringenc package*; 2007-06-16 v1.1; CTAN:macros/latex/contrib/oberdiek/stringenc.pdf.

7 History

[2007/06/16 v1.0]

- First version.

[2007/06/20 v1.1]

- Requested date for package stringenc fixed.

[2007/09/09 v1.2]

- Line end fixed.

[2016/05/16 v1.3]

- Documentation updates.

[2016/05/17 v1.4]

- Documentation updates: Avoid T1 encoding with Unicode TEX.

8 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.

T

U

W