Program and package `xindex`

–

0.06 (January 10, 2019)

Herbert Voß[*]

January 10, 2019

Contents

---

[*]hvoss@tug.org

1 Introduction

The Lua program `xindex` is a unicode aware program for creating an index file from an `.idx` source file. It is completely compatible to the current `makeindex` program, but can handle UTF-8, 16, 32, and 64. The LaTeX package `xindex` is a package which provides a LaTeX command which writes additional text into the index file. This text (comments and/or macros) will be accepted by the program `xindex`.

The general structure of a data element in the Lua table is:

```
data = { Entry = ‹text›,  -- like the input line without command \indexentry
        pages = {
                  { number  = ‹roman/arabic number or text›,
                    special = ‹macro› },  -- the part after | in the input
                  [...]
                  { number  = ‹roman/arabic number or text›,
                    special = ‹macro› }
              },
        sortChar = ‹unicode codepoint›, -- of the first character of Entry
        Macro    = ‹TeX macro›  -- only useful with LaTeX package \Lpack{xindex}
      }
```

After reading the input file the table pages has only one element for the number and the so-called special command. When the pages are compressed the table will collect all pages which refer to the same entry name.

1.1 Syntax

The syntax is `xindex [...] ‹file›` where `[...]` are optional arguments, either in short or long form which, of course, can be mixed:

```
xindex
    [-q,--quiet ]
    [-h,--help ]
    [-v ]                           verbose
    [-c,--config ]                  default is cfg
    [-e,--escapechar ]              default is "
    [-n,--noheadings ]
    [-o,--output ]                  default is ‹input›.ind
    [-l,--language ]                default is en
    ‹input file›
```

The language has to be chosen as an international abbreviation in lower- or uppercase letters, see https://en.wikipedia.org/wiki/ISO_3166-2

1.2 How it works

xindex creates by default an output file ‹input›.ind which can be read by the LATEX document with the default command \printindex. One can use another output filename, which makes only sense if one doesn't use the \printindex command for typesetting the index. The default sorting is given by the configuration file, which defines replacements for accented characters, like ö→o.

1.3 The .idx file

There are four characters which must be escaped if used in the command \index: !, @, ", or |. These charactzers have a special meaning for the index. The default escape character is the double quote ":

```
\usepackage{makeidx}\makeindex
```

```
\section{Escaping characters}
\begin{itemize}
\item Exclamation mark ! \index{exclaim ("!)}
\item Vertical bar| \index{Vertical bar ("|)}
\item Doublequote \verb|"| \index{""}
\item Double doublequote \verb|""| \index{""""}
\item At character @ \index{At ("@)}
\end{itemize}
run \texttt{xindex ‹file.idx›}\index{‹file.idx›}\index{123}
\newpage
\printindex
```

1 Escaping characters

- Exclamation mark !

- Vertical bar|

- Doublequote "

- Double doublequote ""

- At character @

run xindex <file.idx>

Index

Symbols
¨, 1
¨¨, 1
<file.idx>, 1

Numbers
123, 1

A
At (@), 1

E
exclaim (!), 1

V
Vertical bar (|), 1

For the german language the double quote is an active character and it makes live easier if one chooses another character. The escape character can be changed easily by the optional argument –e "‹char›" or ––escapechar "‹char›". The following example shows how it works for the escape character > (greater). Internally the escape sequences are now defined as:

```
escape_chars = { -- by default " is the escape char
  {esc_char..'"', '//escapedquote//', '\\"{}' },
  {esc_char..'@', '//escapedat//',    '@'    },
  {esc_char..'|', '//escapedvert//',  "|"    },
  {esc_char..'!', '//scapedexcl//',   '!'    }
}
```

which is, of course, not of interest for the user. With the beginning the escaped chars are converted into the internal strings and later back to the origin meaning.

```
\usepackage{makeidx}\makeindex
```

```
\section{Escaping characters with ›}
\begin{itemize}
\item Exclamation mark ! \index{exclaim (›!)}
\item Vertical bar| \index{Vertical bar (›|)}
\item Doublequote \verb|"| \index{›"}
\item Double doublequote \verb|""| \index{›">}
\item At character @ \index{At (›@)}
\end{itemize}
Run \texttt{xindex} with \texttt{xindex -e ">"}\index{‹file.idx›}\index{123}
\newpage
\printindex
```

---

1 Escaping characters with ❭

- Exclamation mark !

- Vertical bar|

- Doublequote "

- Double doublequote ""

- At character @

Run xindex with xindex -e ">"

Index

Symbols
¨, 1
¨¨, 1
<file.idx>, 1

Numbers
123, 1

A
At (@), 1

E
exclaim (!), 1

V
Vertical bar (|), 1

---

2 Language

The language is only important for the first two headers in the output of the index data. They are by default *Symbols* followed by *Numbers*. In a new version of xindex it will be customizable. The predefined language is »en« and currently the following languages are possible:

```
indexheader = {
de = { Symbole , Zahlen },
en = { Symbols , Numbers },
fr = { Symboles , Chiffre },
jp = { シンボル , 番号 },
}
```

The following example was run with `xindex -l de ‹file›.idx`:

```
\usepackage{makeidx}\makeindex
```

```
\section{Escaping characters with ›}
\begin{itemize}
\item Exclamation mark ! \index{exclaim (›!)}
\item Vertical bar| \index{Vertical bar (›|)}
\item Doublequote \verb|"| \index{›"}
\item Double doublequote \verb|""| \index{›"›"}
\item At character @ \index{At (›@)}
\end{itemize}
Run \texttt{xindex} with \texttt{xindex -l de -e ">"}\index{123}
\newpage
\printindex
```

1 Escaping characters with >

- Exclamation mark !

- Vertical bar|

- Doublequote "

- Double doublequote ""

- At character @

Run `xindex` with `xindex -l de -e ">"`

Index

Symbole
¨, 1
¨¨, 1

Zahlen
123, 1

A
At (@), 1

E
exclaim (!), 1

V
Vertical bar (|), 1

---

3 Sorting

The default sorting is unicode aware and uses a translation table for accented characters:

```
alphabet_lower = { --   for sorting
    { ' ' },  -- only for internal tests
    { 'a', 'á', 'à', 'ä', 'å', 'æ', },
    { 'b' },
```

```
    { 'c', 'ç' },
    { 'd' },
    { 'e', 'é', 'è', 'ë' },
    { 'f' },
    { 'g' },
    { 'h' },
    { 'i', 'í', 'ì', 'ï' },
    { 'j' },
    { 'k' },
    { 'l' },
    { 'm' },
    { 'n', 'ñ' },
    { 'o', 'ó', 'ò', 'ö', 'ø', 'œ'},
    { 'p' },
    { 'q' },
    { 'r' },
    { 's', 'š', 'ß' },
    { 't' },
    { 'u', 'ú', 'ù', 'ü' },
    { 'v' },
    { 'w' },
    { 'x' },
    { 'y', 'ý', 'ÿ' },
    { 'z', 'ž' }
}
```

There is also a table for the uppercase letters. If it should be edited or extended then copy first the base configuration file `xindex-cfg.lua` and modify that new file. It can be used by `xindex` with the optional argument `-c newfile` if it is named as `xindex-newfile.lua`. For german there already exists a configuration file `xindex-DIN2.lua` which uses the so-called »Telefonbuchsortierung« which converts the umlauts like ö→oe:

```
alphabet_upper = { -- for sorting
    { ' ' },
    { 'A', 'Á', 'À', 'Å', 'Æ'},
    { 'AE', 'Ä'},
    { 'B' },
    { 'C', 'Ç' },
    { 'D' },
    { 'E', 'È', 'È', 'Ë' },
    { 'F' },
    { 'G' },
    { 'H' },
    { 'I', 'Í', 'Ì', 'Ï' },
    { 'J' },
    { 'K' },
    { 'L' },
```

```
    { 'M' },
    { 'N', 'Ñ' },
    { 'O', 'Ó', 'Ò', 'Ø','Œ' },
    { 'OE', 'Ö' },
    { 'P' },
    { 'Q' },
    { 'R' },
    { 'S', 'Š' },
    { 'T' },
    { 'U', 'Ú', 'Ù' },
    { 'UE', 'Ü' },
    { 'V' },
    { 'W' },
    { 'X' },
    { 'Y', 'Ý', 'Ÿ' },
    { 'Z', 'Ž' }
}
```

```
\usepackage{makeidx}\makeindex
\newcommand\Index[1]{\index{#1}#1}


Sorted with \verb|-1 DE|

\Index{Österreich} \Index{Öresund}
\Index{Ostern} \Index{Ober}
\Index{Oberin} \Index{Österreich}
\Index{Öresund} \Index{Ostern}
\Index{Ober} \Index{Oberin}
\Index{Obstler} \Index{Öl}
\Index{ölen} \Index{Ödem}
\Index{Oligarch} \Index{Oder}
\Index{oder} \index{Fluss!Oder}
\index{Oder|seealso{Fluss}}
\Index{Göbel} \Index{Goethe}
\Index{Göthe} \Index{Götz}
\Index{Goldmann}

\printindex
```

Index

F
Fluss
      - Oder, 1
G
Göbel, 1
Goethe, 1
Goldmann, 1
Göthe, 1
Götz, 1
O
Ober, 1

Oberin, 1
Obstler, 1
Ödem, 1
oder, 1
Oder, 1, *siehe auch* Fluss
Öl, 1
ölen, 1
Oligarch, 1
Öresund, 1
Ostern, 1
Österreich, 1

xindex–4

The same sorted with the german DIN variant 2 with --config DIN2, which is part of the TEX distribution. In this case a letter Ö is converted to Oe before sorting the word beginninjg with the letter Ö:

```
\usepackage{makeidx}\makeindex
\newcommand\Index[1]{\index{#1}#1}

Sorted with
\verb|--config DIN2 -l DE|

\Index{Österreich} \Index{Öresund}
\Index{Ostern} \Index{Ober}
\Index{Oberin} \Index{Österreich}
\Index{Öresund} \Index{Ostern}
\Index{Ober} \Index{Oberin}
\Index{Obstler} \Index{Öl}
\Index{ölen} \Index{Ödem}
\Index{Oligarch} \Index{Oder}
\Index{oder} \index{Fluss!Oder}
\index{Oder|seealso{Fluss}}
\Index{Göbel} \Index{Goethe}
\Index{Göthe} \Index{Götz}
\Index{Goldmann}

\printindex
```

Index

F
Fluss
    - Oder, 1
G
Göbel, 1
Goethe, 1
Göthe, 1
Götz, 1
Goldmann, 1
O
Ober, 1

Oberin, 1
Obstler, 1
oder, 1
Oder, 1, *siehe auch* Fluss
Ödem, 1
Öl, 1
ölen, 1
Öresund, 1
Österreich, 1
Oligarch, 1
Ostern, 1

The following runs with xindex -l jp ‹file›:

```
\usepackage{fontspec}
\setmainfont{SourceHanSans}
\usepackage[japanese]{babel}
\addto\captionsjapanese{%
 \def\indexname{指数}}
\usepackage{hvindex}% for \Index
\usepackage{makeidx}\makeindex
```

```
\Index{車} \Index{車道}
日本\index{日本|fbox}
\Index{病院} \Index{コンピュータ}
\Index{プリンタ} \Index{印刷}
\Index{スイミングプール} \Index{天王}
\Index{広島} \Index{ドイツ}
\Index{日本} \Index{ワープロ}
\Index{foo} und \Index{bar}
\Index{//} \Index{4711}
\newpage\printindex
```

指数

シンボル
//, 1

番号
4711, 1

B
bar, 1

F
foo, 1

コ
コンピュータ, 1

ス
スイミングプール, 1

ド
ドイツ, 1

プ
プリンタ, 1

ワ
ワープロ, 1

印
印刷, 1

天
天王, 1

広
広島, 1

日
日本, 1, ☐1

病
病院, 1

車
車, 1
車道, 1

4 Compressing pagenumbers

By default page sequences of an entry are compressed to

8f page 8 and 9

8ff page 8, 9, and 10

8-12 page 8, 9, ..., 12

The so-called folio abbreviation is language dependent and defined in the file `xindex-cfg-common.cfg`:

```
folium = {
 de = {"f", "ff"},
 en = {"f", "ff"},
 fr = {"\,sq","\,sqq"},
 jp = {"シンボル","番号"},
}
```

```
\usepackage{makeidx}\makeindex
```

```
Sorted with \verb|-l fr|

foobar\index{foobar|(}
foo\index{foo}\index{bar}\index{baz}\newpage
foo\index{foo}\index{bar}\index{baz}\newpage
foo\index{bar}\index{baz}\newpage
foo\index{baz}\newpage
foo\index{foo}foobar\index{foobar|)}
\newpage
\printindex
```

Index

B
bar, 1 sqq
baz, 1–4

F
foo, 1 sq, 5
foobar, 1–5

5 `hyperref`

Using the package `hyperref` is no problem:

```
\usepackage{makeidx}\makeindex
\usepackage{hvindex}% for \Index
\usepackage[colorlinks]{hyperref}
```

```
Sorted with \verb|-l DE|
\Index{Österreich} \Index{Öresund}
\Index{Ostern} \Index{Ober} \Index{Oberin}
\Index{Österreich} \index{Öresund|textbf}
\Index{Ostern} \Index{Ober} \Index{Oberin}
\Index{Obstler} \Index{Öl} \Index{ölen}
\Index{Ödem} \Index{Oligarch} \Index{Oder}
\Index{oder} \index{Fluss!Oder|textit}
\Index{Oder|seealso{Fluss}} \Index{Göbel}
\Index{Goethe} \Index{Göthe} \Index{Götz}
\newpage\Index{Goldmann} \Index{Goethe}
\newpage \printindex
```

Index

F
Fluss
    - Oder, *1*
G
Göbel, 1
Goethe, 1f
Goldmann, 2
Göthe, 1
Götz, 1
O
Ober, 1

Oberin, 1
Obstler, 1
Ödem, 1
Oder, 1
oder, 1
Oder, *see also* Fluss
Öl, 1
ölen, 1
Oligarch, 1
Öresund, 1, **1**
Ostern, 1
Österreich, 1

6 Page argument

Every page can be combined with an additional macro, like \index{foo|fbox}, the page number will be set into a framebox. If we have on the same page the two commands:

foo\index{foo} and foo\index{foo|bar}

then we have two *different* index entries which will not be compressed to one entry. In the follwoing example we have four different entries for *foo* which is the reason that we do not get an output like foo, 1--4. Only the first two entries are of the same type, so we get 1f in the output.

```
\usepackage{makeidx}\makeindex

Ein foo\index{foo} \newpage und \index{foo}
ein foo\index{foo|textit} \newpage
und foo\index{foo|textbf} \newpage
und foo\index{foo|fbox}

\newpage
\printindex
```

xindex-9

Index

F
foo, 1f, *2*, **3**, 4

7 The config file

The main config file is xindex-cfg.lua is used by default and loading it by the optional parameter -c makes no sense. A new config file must have the prefix xindex- and the file extension .lua, for example: xindex-HAdW-eKO.lua which can be used with --config HAdW-eKO. The file must be saved in the documents directory or in one which is known to kpsewhich, for example[1] $TEXMFLOCAL/tex/lualatex/xindex/ Don not forgot to update the filename database.

A new config file must declare at least the variables which are part of the default config file: the translation tables and

```
itemPageDelimiter = ","      -- Hello, 14
compressPages     = true
  -- something like 12--15, instaead of 12,13,14,15. the |( ... |) syntax is still valid
fCompress   = true    -- 3f -› page 3, 4 and 3ff -› page 3, 4, 5
minCompress       = 3        -- 14--17 or
numericPage       = true     -- for non numerical pagenumbers, like "VI-17"
sublabels         = {"", "-\\-", "--\\-", "---\\-"}
   -- for the sub(sub(sub-items, first one is empty
pageNoPrefixDel   = ""       -- a delimiter for page numbers like "VI-17"  -- not used !!!
indexOpening      = ""       -- commands/text after \begin{theindex}
```

---

[1]The directory xindex must be created before saving the file.

The new config file can define own functions for compressing the pagelist for a given entry and for the formatting of the output. They must be called `specialCompressPageList` and `specialGetPageList`.

For example:

```
function specialCompressPageList(pages)
  if (pages[1]["number"] == "") then pages[1]["number"] = " " end
  if (#pages <= 1) then
    pages[1]["number"] = pages[1]["number"]:gsub('(.-)-',colorBox)-- replace "-" with ":\\ "
    return pages
  end  -- only one pageno
  local sortPages = {}
  local roman
  local volume
  local page
  local i
  for i=1,#pages do
    roman = string.gsub(pages[i]["number"],'%U*','') -- only uppercase to catch VII123f (folium pages)
    if romanToNumber(roman) then
      roman = string.format("%05d",tonumber(romanToNumber(roman)))
    else
      roman = ""
    end
    volume = string.gsub(pages[i]["number"],'%a*','')
    if volume then volume = volume:gsub('-%d*','') end
    page = string.gsub(pages[i]["number"],'.*-','')
    page = string.format("%5s",page)
    sortPages[#sortPages+1] = {
    origin = pages[i],
    sort = roman..volume.." "..page }  -- no minus between Roman/Volume and first page
  end
-- sort the page list
  table.sort(sortPages, function(a,b) return a["sort"] < b["sort"] end )
  local Pages = {}
  for i=1,#sortPages do     -- use the sorted origin table
    Pages[#Pages+1] = sortPages[i]["origin"]
  end

[...]

return pages
end
```

is a special function which can handle page numbers like VII-17, VIII/2/1-186. Internally exists a function `compressPageList` which is used if no `specialCompressPageList` is defined.

```
\usepackage{makeidx}
```

```
\mbox{}\printindex
```

Personenverzeichnis

The config file `xindex-dtk.lua` defines a special page output:

```lua
function specialGetPageList(v,hyperpage)  -- Entry table, boolean
  local Pages = {}
[..]
      if (Pages[1]["special"] == nil) or (Pages[1]["number"] == nil) then return ""  end
      if #Pages == 1 then
        return "\\relax"..Pages[1]["number"].."\\@nil"
      else
        pageNo = "\\relax"..Pages[1]["number"]
        for i=2,#Pages do
          if Pages[i]["number"] then
            pageNo = pageNo..", "..Pages[i]["number"].."\\@nil"
            Pages[i] = {}
          end
        end
[..]
end
```

The following example runs `xindex -c dtk -l de -n ‹input›`

```
\usepackage{makeidx}
```

```
\mbox{}\printindex
```

Autorenliste

Doris Baehrendtsen	[14]	**Eike Schuster**	[40]
dori@xyz.de		Haussteig 15	
		36396 Stuttens	
		elke.schuster@kabelxyz.de	
Jürgen Fannt	[48]		
Friedrichallee 74		**Martin Severs**	[4]
13233 Neu-Isenburg		siehe Seite **??**	
juergen.fannt@gmxnet.de			
		Herbert Voß	[3, 5]
Jonasson Jared Jazek	[20]	Wasgenstraße 121	
mail@jones.net		10127 Potsdam	
		herbert@xyz.de	
Martin Koomer	[24, 31]	**Uwe Ziegendaten**	[9]
Freiherr-von-Stein-Weg 16		Lokostr. 19	
15525 Erdingen-Neckar		20713 Kalln	
kooma@xyz.info		ziegendaten@mail.com	

7.1 Sublabels

There are three predefined sublabels for \subitems. The program itself can handle more, there is no limit for xindex.

```
\makeatletter
\g@addto@macro{\theindex}{%
  \pagestyle{empty}%
  \def\subsubsubitem{\@idxitem\hspace*{35\p@}}
  \def\subsubsubsubitem{\@idxitem\hspace*{40\p@}}
}
\makeatother
\usepackage{makeidx}\makeindex
```

```
foo\index{foo} bar\index{foo!bar}
baz\index{foo!bar!baz} foobar%
\index{foo!bar!baz!foobar} Kuba
\index{foo!bar!baz!foobar!Kuba}
\newpage \printindex
```

Index

B
bar, 1

F
foo, 1

G
gex, 1

8 Including LaTeX commands into the `.idx` file

The command \addtocontents doesn't work for the index file. With the LaTeX package xindex (same name as the Lua program xindex) defines a macro \writeidx which writes its argument into the .idx file. This can be usefull to insert a pagebreak/""columnbreak before a new letter in the output of the index file:

```
\documentclass{article}
\usepackage{makeidx}
\makeindex
```

```
\usepackage{xindex}
\begin{document}

\index{foo}foo and
\writeidx{\clearpage}
\index{bar}bar

\printindex
\end{document}
```

Such commands are then taken into account by the program xindex. With the often used program makeindex such commands are ignored. In the following example we put an horizontal line after the first entry:

xindex-13

```
\usepackage{xindex}
\makeindex

\index{foo}foo and
\writeidx{\item\protect\hrulefill}
\index{bar}bar
\index{gex}gex
\printindex
```

Index

B
bar, 1

F
foo, 1

G
gex, 1

9 Headings

By default the output uses the english headings: *Symbols*, *Numbers*, and *A* ...There are three predefined languages en, de, and fr. The definition is in the file xindex-cfg-common.lua (see also section 2 on page 4). It can easily be extended for other languages. Sometimes the headers are not needed, for example in a name list. With the optional argument -n or --noheadings the created .ind file has only the vertical space between different first letters:

xindex-14

```
\usepackage{makeidx}\makeindex
```

```
Ein foo\index{foo}\index{bar|(}
 \newpage und \index{foo}
ein foo\index{foo|textit} \newpage
und foo\index{foo|textbf} \newpage
und foo\index{foo|fbox}
\index{bar|)}
\newpage
\verb|xindex -n <file>|
\printindex
```

```
xindex -n <file>
```

Index

bar, 1–4

foo, 1f, *2*, **3**, ⏥4

10 Automatic index creation

With package xindex one can define several different index files, e. g. an index of names. With the optional argument imakeidx the package itself loads imakeidx and adds the program xindex as the default program to imakeidx.

```
\usepackage[imakeidx]{xindex}
\makeindex[name=persons,title=Index of names,
   columns=1,options=--noheadings]
\def\ThanhVN{Hàn Thê\protect\llap{%
  \raise 0.5ex\hbox{\'{}}}}
```

```
foo\index[persons]{Niepraschk,~ Rolf}
foo\index[persons]{Lamport,~ Leslie}
foo\index[persons]{Knuth,~ Donald}
foo\index[persons]{Knuth,~ Donald}
\newpage
foo\index[persons]{Lamport,~ Leslie}
foo\index[persons]{Thành,~ \ThanhVN}
foo\index[persons]{Kew,~ Jonathan}
foo\index[persons]{Kohm,~ Markus}
foo\index[persons]{Preining,~ Norbert}
\newpage
foo\index[persons]{Schenk,~ Christian}
foo\index[persons]{Feuerstack,~ Thomas}
foo\index[persons]{Tobin,~ Geoffrey}
foo\index[persons]{Wilson,~ Peter}
\newpage
foo\index[persons]{Kohm,~ Markus}
foo\index[persons]{Theiling,~ Henrik}
foo\index[persons]{Pégourié-Gonnard,~ Manuel}
foo\index[persons]{Roux,~ Élie}
\newpage
foo\index[persons]{Mittelbach,~ Frank}
foo\index[persons]{Fairbairns,~ Robin}
foo\index[persons]{Lemberg,~ Werner}
foo\index[persons]{Volovich,~ Vladimir}

\printindex[persons]
```

Index of names

Fairbairns, Robin,
Feuerstack, Thomas,

Kew, Jonathan,
Knuth, Donald,
Kohm, Markus,

Lamport, Leslie,
Lemberg, Werner,

Mittelbach, Frank,

Niepraschk, Rolf,

Pégourié-Gonnard, Manuel,
Preining, Norbert,

Roux, Élie,

Schenk, Christian,

Thành, Hàn Thế,
Theiling, Henrik,
Tobin, Geoffrey,

Volovich, Vladimir,

Wilson, Peter,

You have to run LATEX with the --shell-escape option to run xindex from within the LATEX document.

11 Demerits

- For more than 5000 entries in the .idx file the internal Lua function for sorting may take some time.

- The .idx file is not checked for LATEX errors in the argument of \indexentry.

Index