

Program and package xindex

—

0.46 (January 18, 2023)

Herbert Voß

hvoss@tug.org

Contents

1. Introduction	2
1.1. Syntax	2
1.2. How it works	3
1.3. The .idx file	3
2. Language	5
3. Sorting	7
3.1. Default sorting by a character table	7
3.2. Sorting by using UCA (Unicode Collation Algorithm)	10
3.3. Case sensitive index entries	11
3.4. Ignore space for sorting	12
3.5. Compressing pagenumber series	12
3.6. Modify Pagenumber	13
4. The config file	14
5. hyperref	17
6. Including L^AT_EX commands into the .idx file	17
7. Headings	18
8. Automatic index creation	19
9. Labels	20
10. Demerits	21
A. Examples	21

Thanks to:

Mark Baudoin; Denis Bitouzé; Andreas Deininger; Brian Dunn; Gonzalo Pesquero Serrano; Michal Hoftich; Mykel Kochenderfer; Frank Mittelbach; Peter Müller; Heiko Oberdiek; Matteo Paolini; Stefan Pinnow; Florent Rougon; Martin Sievers; Simon Spiegel

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(s)>` 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 "
    [-a, --no_casesensitive ] default is false
    [-b,--no_labels ] default is false
    [-n,--noheadings ] default is false
    [-i,--ignoreSpace ] default is false
    [-o,--output ] default is <input>.ind
    [-k,--checklang ] default ist false
    [-l,--language ] default is en
    [-p,--prefix] default L
    [-u,--use_UCA ] default is false
    [-s,--use_stdin ] default is false
    [-V,--version ]
    <files...> (default stdin) file(s)[.idx] one or more files
```

For example:

```
xindex -q -l fr -b myDoc
xindex -l de -c DIN2 demo1 demo2 demo3
xindex -c norsk -o index.ind demo1 demo2.bdx demo3.adx
```

1. -q: quiet; -l fr: french language setting; -b: no labels; myDoc: input data myDoc or myDoc.idx)
output data will be myDoc.ind and logfile myDoc.ilg
2. -l de: German language setting; -c DIN2: config file xindex-DIN2.lua; demo1 demo2 demo3: input
data files with or without extension .idx
output data will be demo1.ind and logfile xindex.ilg
3. -c norsk: config file xindex-norsk.lua; -o index.ind: output file; demo1 demo2.bdx demo3.adx: in-
put data files with or without extension .idx
output data will be xindex.ind and logfile xindex.ilg

It is also possible to use standard input for the index data, which needs the -s parameter:

```
cat myDoc.idx | xindex -q -l fr -b -s
xindex -l de -c DIN2 < myDoc.idx
```

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 L^AT_EX 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 characters 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
```

xindex-1.tex

\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 life 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.

xindex-2.tex

\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 which its alias are possible:

```
indexheader = {
  cs = {"Symboly", "Čísła", "czech"},
  da = {"Symboler", "Tal", "danish"},
  de = {"Symbole", "Zahlen", "austrian", "german", "germanb", "ngerman", "naustrian"},
  en = {"Symbols", "Numbers", "english", "USenglish", "american", "UKenglish", "british", "canadian", "australian", "australianb", "naustralian"},
  es = {"Símbolos", "Números", "spanish"},
  fr = {"Symboles", "Nombres", "french", "français", "canadien", "acadian"},
  it = {"Simboli", "Numeri", "italian"},
  jp = {"シンボル", "番号", "japanese"},
  nl = {"Symbolen", "Nummers", "dutch"},
  no = {"Symboler", "Tall", "norsk", "nynorsk"},
}
```

The following example was run with `xindex -l it <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}
```

xindex-3.tex

```
Run \texttt{xindex} with \texttt{xindex -l it -e ">"}\index{123}
\newpage
\printindex
```

1 Escaping characters with >

- Exclamation mark !
- Vertical bar|
- Doublequote "
- Double doublequote ""
- At character @

Run xindex with xindex -l it -e ">"

Indice analitico

Simboli

", 1
"", 1

Numeri

123, 1

A

At (@), 1

E

exclaim (!), 1

V

Vertical bar (|), 1

The following example was run with `xindex -k <file>.idx`. In this case `xindex` tries to detect the language from the aux file(s). This is only possible if package `babel` or `polyglossia` are used.

xindex-4.tex

```
\usepackage[dutch]{babel}
\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 "*" -e "?"}\index{123}
\newpage
\printindex
```

1 Escaping characters with ?

- Exclamation mark !
- Vertical bar|
- Doublequote "
- Double doublequote ""
- At character @

Run xindex with `xindex -l "*-e "?"`

Index

Symbole

", 1
, 1

Nummers

123, 1

A

At (@), 1

E

exclaim (!), 1

V

Vertical bar (|), 1

3. Sorting

3.1. Default sorting by a character table

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' },
  { 'b' },
  { 'c' },
  { 'd' },
  { 'e', 'é', 'è', 'ë' },
  { 'f' },
  { 'g' },
  { 'h' },
  { 'i', 'í', 'ì', 'ï' },
  { 'j' },
  { 'k' },
  { 'l' },
  { 'm' },
  { 'n', 'ñ' },
  { 'o', 'ó', 'ò', 'ö' },
  { 'p' },
  { 'q' },
  { 'r' },
  { 's' },
  { 't' },
  { 'u', 'ú', 'ù', 'ü' },
  { 'v' },
```

3. Sorting

```
{ 'w' },  
{ 'x' },  
{ 'y' },  
{ 'z' }  
}
```

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', 'Ä' },  
  { 'B' },  
  { 'C' },  
  { 'D' },  
  { 'E', 'È', 'É', 'ë' },  
  { 'F' },  
  { 'G' },  
  { 'H' },  
  { 'I', 'Í', 'Ì', 'ï' },  
  { 'J' },  
  { 'K' },  
  { 'L' },  
  { 'M' },  
  { 'N', 'Ñ' },  
  { 'O', 'Ó', 'Ò', 'Ö' },  
  { 'P' },  
  { 'Q' },  
  { 'R' },  
  { 'S' },  
  { 'T' },  
  { 'U', 'Ú', 'Ù', 'Ü' },  
  { 'V' },  
  { 'W' },  
  { 'X' },  
  { 'Y' },  
  { 'Z' }  
}
```



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

Sorted with `\verb|-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	Oberin, 1
- Oder, 1	Obstler, 1
G	Ödem, 1
Göbel, 1	Oder, 1
Goethe, 1	oder, 1
Goldmann, 1	Oder, <i>siehe auch</i> Fluss
Göthe, 1	Öl, 1
Götz, 1	ölen, 1
O	Oligarch, 1
Ober, 1	Öresund, 1
	Ostern, 1
	Österreich, 1

xindex-5.tex

The same sorted with the German DIN variant 2 with `--config DIN2`, which is part of the \TeX distribution. In this case a letter \ddot{O} is converted to Oe before sorting the word beginning with the letter \ddot{O} :

```
\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	Öl, 1
- Oder, 1	ölen, 1
G	Öresund, 1
Göbel, 1	Österreich, 1
Göthe, 1	Ober, 1
Goethe, 1	Oberin, 1
Götz, 1	Obstler, 1
Goldmann, 1	oder, 1
O	Oder, 1, <i>siehe auch</i> Fluss
Ödem, 1	Oligarch, 1
	Ostern, 1

xindex-6.tex

The following runs with `xindex -l jp <file>`:

```
\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
```

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

xindex-7.tex

3.2. Sorting by using UCA (Unicode Collation Algorithm)

With the optional argument `-u` or alternatively `--use_UCA` the sorting will be done by Michal Hoftich's Lua package `LUA-UCA`, which should be part of any \TeX installation.

```
\usepackage{multicol}
\usepackage{makeidx}\makeindex
\def\Index#1{#1\index{#1}}
```

```
Sorted with \verb|-u -l cs|
\Index{ahoj} \Index{crha}, \Index{čaj}, \Index{chachar},
\Index{rak}, \Index{řeka}, \Index{srp}, \Index{šutr},
\Index{hudba}, \Index{linux}, \Index{zebra},
\Index{žába}, \Index{7 dubů}
\begin{multicols}{2} \printindex \end{multicols}
```

Sorted with `-u -l cs` ahoj crha, čaj, chachar, rak, řeka, srp, šutr, hudba, linux, zebra, žába, 7 dubů

Index	
A	L
ahoj, 1	linux, 1
C	R
crha, 1	rak, 1
Č	Ř
čaj, 1	řeka, 1
H	S
hudba, 1	srp, 1
Ch	šutr, 1
chachar, 1	Z
	žába, 1
	zebra, 1
	Čísla
	7 dubů, 1

xindex-8.tex

The sorting order can be easily modified. Read the documentation of the package `LUA-UCA` on how to do it and what languages are supported so far. Any additional code setting for UCA should be don't in the file `xindex-cfg-uca.lua`, which will automatically be read by `xindex`.

3.3. Case sensitive index entries

By default `foo` and `Foo` are two different entries and will handled differently by `xindex`: `Foo` will be as an own entry *before* `foo`. Let's see a more complex example. In the index the entry `xindex-DIN2.lua` is the first one of the `xindex-???` series because uppercase letters are sorted before lowercase letters.

```
\usepackage{makeidx}
\usepackage{hyperref}
```

```
foo\newpage
\printindex
```

Index

X
xindex package, 2, 15
xindex program, 4, 13f
xindex-DIN2.lua file, 6
xindex-HAdW-eK0.lua file, 10
xindex-cfg-common.cfg file, 9
xindex-cfg-common.lua file, 14
xindex-cfg.lua file, 6, 10
xindex-dtk.lua file, 12
xindex-newfile.lua file, 6

xindex-9.tex

3. Sorting

The same example sorted with the `-a` or `--no_casesensitive` has another output: now `xindex-cfg-common.lua` is the first one of the `xindex-???` series.

xindex-10.tex

```
\usepackage{makeidx}
\usepackage{hyperref}
```

```
foo\newpage
\printindex
```

Index

X

xindex package, 2, 15
xindex program, 4, 13f
xindex-cfg-common.cfg file, 9
xindex-cfg-common.lua file, 14
xindex-cfg.lua file, 6, 10
xindex-DIN2.lua file, 6
xindex-dtk.lua file, 12
xindex-HAdW-eKO.lua file, 10
xindex-newfile.lua file, 6

3.4. Ignore space for sorting

By default alpha sort will be sorted *before* alphaA:

xindex-11.tex

```
\usepackage{makeidx}\makeindex
% default sorting
```

```
Test
\index{alpha sort}\index{alphaA}
\newpage
\printindex
```

Index

A

alpha sort, 1
alphaA, 1

This can be changed with the optional argument `-i` or `--ignoreSpace`:

xindex-12.tex

```
\usepackage{makeidx}\makeindex
% sort with xindex -i <file>
```

```
Test
\index{alpha sort}\index{alphaA}
\newpage
\printindex
```

Index

A

alphaA, 1
alpha sort, 1

3.5. Compressing pagenumber series

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 = {
  cs = {"f.", "ff."},
  da = {"f", "ff"},
  de = {"f", "ff"},
  en = {"f", "ff"},
  es = {"f", "ff"},
  fr = {"\\,sq.", "\\,sqq."},
  it = {"f", "ff"},
  jp = {"シンボル", "番号"},
  no = {"\\,f.", "\\,ff."},
}
```

```
\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

xindex-13.tex

3.6. Modify Pagenumber

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 following 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
```

Index

F

foo, 1f, 2, 3, 4

xindex-14.tex

4. 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-eK0.lua` which can be used with `--config HAdW-eK0`. The file must be saved in the documents directory or in one which is known to `kpsewhich`, for example¹ `$TEXMFLOCAL/tex/lualatex/xindex/` Do not forget 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, instead 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
rangeSymbol         = "--"
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}
```

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('-',':~')-- 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 VII/1-123f and VII/3-123ff (folium pages)
    if romanToNumber(roman) then
      roman = string.format("%05d",tonumber(romanToNumber(roman))) -- only roman part VII
    else
      roman = ""
    end
    volume = string.gsub(pages[i]["number"],'%a*', '') -- only the number /2 123 or /2-123
```

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

```

if volume then volume = volume:gsub('-%d*','') end -- delete - char to get /2
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
table.sort(sortPages, function(a,b) return a["sort"] < b["sort"] end )

[...]

return pages
end
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
```

xindex-15.tex

Personenverzeichnis

A

Aachen, Johannes von VII/1 : 215

Aarones VII/2/1 : 1003, 1012

Abrahamson VII/2/1 : 864, 991, 1048, 1067, 1156

Adamson VII/2/1 : 1223, IX/1 : 1228

Adrian

- Hauster VII/1 : 514, XI/1 : 515

Alting

- Mensa VII/1 : 426, 434, 453, 455, 466f.

B

Braunschweig-Wolfenbüttel

- Karl Viktor von, Herzog VI/1 : 83

Bremen

- Heinz von, Erzbischof see Sachsen-Lauenburg

J

Julian

- Apostata, römischer Kaiser VII/2/1 : 904

Justinian I., byzantinischer Kaiser VII/1 : 326, 734, VII/2/1 : 1011

K

Karl

- II., Kaiser VII/1 : 147

- III., Kaiser VII/1 : 149

- IV., Kaiser VI/1 : 12, VII/1 : 34, 147

- V., Kaiser VI/1 : 84, 284, 654, VI/2 : 708, 1014, 1043, 1131, 1210, VII/1 : 34

- VI., Kaiser VII/1 : 296

- IX., Kaiser VII/1 : 296

- X., Kaiser VII/1 : 149

- der Große, Kaiser VI/2 : 987, 989, 1028

O

Osnabrück

- Heinz von, Bischof see Sachsen-Lauenburg

S

Schleswig-Holstein

- Rudolf von, Herzog VII/2/1 : 758–761, 765

Z

Zwingl, Haldrich IX : 479, 692

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

```

function specialGetPageList(v,hyperpage) -- Entry table, boolean
  local Pages = {}

```

4. The config file

```
[...]
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>`

xindex-16.tex

```
\usepackage{makeidx}
```

```
\mbox{}\label{president}
\printindex
```

Autorenliste

Elke Bährendtsen elke@xyz.de	[14]	Eike Schuler Haussteig 15 36396 Stuttgart eike.schulter@kabel.de	[40]
Horst Fannt Friedrichallee 74 13233 Neu-Isenburg horst.fannt@gmxnet.de	[48]	Markus Severs siehe Seite 1	[4]
Jonasson Jared Jazek mail@jones.net	[20]	Herbert Voß Wasgensteig 12 10127 Potsdam herbert@xyz.de	[3, 5]
Martin Koon Freiherr-Links-Weg 16 15525 Neckar koo@xyz.org	[24, 31]	Michael Ziegenda Lokostr. 19 20713 Kallin ziegenda@mail.com	[9]

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

xindex-17.tex

```
\makeatletter
\g@addto@macro{\theindex}{%
  \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

F
foo, 1
- bar, 1
- baz, 1
- foobar, 1
- Kuba, 1

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	Oberin, 1
- Oder, 1	Obstler, 1
	Ödem, 1
	Oder, 1
G	oder, 1
Göbel, 1	Oder, <i>see also</i> Fluss
Goethe, 1f	Öl, 1
Goldmann, 2	ölen, 1
Göthe, 1	Oligarch, 1
Götz, 1	Öresund, 1, 1
O	Ostern, 1
Ober, 1	Österreich, 1

xindex-18.tex

6. Including L^AT_EX commands into the .idx file

The command `\addtocontents` doesn't work for the index file. With the L^AT_EX package `xindex` (same name as the Lua program `xindex`) defines a macro `\writeidx` which writes its argument into the .idx file. This can be useful 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-19.tex

```
\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

7. 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 5). 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-20.tex

```
\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

The headings are printed by default as \textbf. This can be changed in the config file by setting the variable idxnewletter, for example: idxnewletter = "\\textit". If you need some more code here then define an own macro for it, which can be seen in the following example. It has an own config file xindex-header.lua which has the line

```
idxnewletter = "\\idxnewletter"
```

In the documents preamble there is the definition:

```
\newcommand\idxnewletter[1]{\textbf{\textit{#1}}}
```

```

æšžŒŠŸŽ
\usepackage{makeidx}\makeindex
\newcommand\idxnewletter[1]{\textbf{\textit{#1}}}

```

```

æšžŒŠŸŽ
\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 \verb|xindex -c header <file.idx>|
\index{<file.idx>@\texttt{<file.idx>}}
\index{123}
\newpage
\printindex

```

Index

Symbols

”, 1
 ””, 1
 <file.idx>, 1

Numbers

123, 1

A

At (@), 1

E

exclaim (!), 1

V

Vertical bar (|), 1

xindex-21.tex

8. 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.

xindex-22.tex

```

\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 Thê,
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.

9. Labels

By default `xindex` creates labels in the index for the symbols, numbers, and other parts (letters) to which one can refer. with `\ref{label}`. The labels are named `L-xindex-<name>`. The prefix `L` can be changed by the config file. `<name>` maybe symbols, numbers, or `A` (a letter). For example

```

\begin{theindex}
\par\textbf{Symbols}\label{L-xindex-symbols}
\nopagebreak[4]
  \item @, \hyperpage{3}
  \item (, \hyperpage{3}
  \item !, \hyperpage{3}

\indexspace

```

```
\textbf{A}\label{L-xindex-A}
[...]
```

The labels can be used to create a reference to a specific part in the index, for example the letter X is in the index on page 23 (\pageref{L-xindex-X}).

With the optional argument -b for the run of xindex one can suppress the creation of the labels, e.g. xindex -b -l fr ...

10. 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 L^AT_EX errors in the argument of \indexentry.

A. Examples

```
\usepackage[imakeidx]{xindex}
\makeindex
% Brian Dunn
```

```
First level.\index{first level}

First level second level.\index{first level!second level}

Duplicate.\index{first level!second level}

Alpha.\index{alpha}

Alpha beta.\index{alpha!beta}

Alpha beta gamma.\index{alpha!beta!gamma}

Duplicate alpha beta.\index{alpha!beta}

Duplicate alpha beta gamma.\index{alpha!beta!gamma}

\newpage
\printindex
```

Index

```
A
alpha, 1
  - beta, 1
    - gamma, 1

F
first level, 1
  - second level, 1
```

xindex-23.tex

```
\usepackage[imakeidx]{xindex}
\makeindex
% Martin Sievers
```

```
Test \index{A!Test} oder auch \index{B!Test}
\newpage
\printindex
```

Index

```
A
A
  - Test, 1

B
B
  - Test, 1
```

xindex-24.tex

xindex-25.tex

```
%% Denis Bitouzé  
\usepackage{makeidx}\makeindex
```

```
Foo\index{foo!bar1!baz1}  
Foo\index{foo!bar1!baz2}  
Foo\index{foo!bar2!baz1}  
Foo\index{foo!bar2!baz2}  
\printindex
```

Index

F

foo

- bar1
 - baz1, 1
 - baz2, 1
- bar2
 - baz1, 1
 - baz2, 1

Index

Symbols

, 3

@, 3

!, 3

A

accented characters, 3

\addtocontents, 17

argument, 2

aux file, 6

B

babel package, 6

C

columnbreak, 17

config file, 15

D

data element, 2

E

entry name, 2

escape character, 3f

H

hyperref package, 17

I

.idx file extension, 2, 17, 21

imakeidx package option, 19

imakeidx package, 19

.ind file extension, 18

\index, 3

index of names, 19

\indexentry, 21

K

kpsewhich program, 14

L

label, 20

language, 3, 5, 18

L^AT_EX errors, 21

.lua file extension, 14

M

makeindex program, 2, 17

N

numbers, 20

O

output, 3

P

page number, 2, 15

pagebreak, 17

\pageref, 21

polyglossia package, 6

\printindex, 3

R

\ref, 20

S

Shell escape, 20

sorting, 3, 21

\subitems, 16

symbols, 20

syntax, 2

T

\textbf, 18

U

unicode, 2

UTF-8, 2

W

\writeidx, 17

X

xindex package, 2, 17, 19

xindex program, 5, 11, 16f, 19f

xindex-cfg-common.cfg file, 13

xindex-cfg-common.lua file, 18

xindex-cfg-uca.lua file, 11

xindex-cfg.lua file, 8, 14

xindex-DIN2.lua file, 8

xindex-dtk.lua file, 15

xindex-HAdW-eK0.lua file, 14

xindex-header.lua file, 18

xindex-newfile.lua file, 8