The pagesel package

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

2016/05/16 v1.9

Abstract

Single pages or page areas can be selected for output.

Contents

1 Usage **2**
1.1 Page selecting . . . 2
1.1.1 Options for selecting pages . . . 2
1.1.2 Examples . . . 2
1.2 Auxiliary files . . . 2
1.2.1 Options for handling auxiliary files . . . 3
1.2.2 Package hyperref . . . 3

2 Implementation **3**
2.1 Package . . . 3
2.2 AtBeginDvi hook support . . . 6

3 Installation **7**
3.1 Download . . . 7
3.2 Bundle installation . . . 7
3.3 Package installation . . . 8
3.4 Refresh file name databases . . . 8
3.5 Some details for the interested . . . 8

4 Catalogue **9**

5 History **9**
[1999/03/01 v0.9] . . . 9
[1999/04/05 v1.0] . . . 9
[1999/04/13 v1.1] . . . 9
[2003/06/05 v1.2] . . . 10
[2006/02/20 v1.3] . . . 10
[2006/03/02 v1.4] . . . 10
[2006/03/07 v1.5] . . . 10
[2007/04/11 v1.6] . . . 10
[2007/04/12 v1.7] . . . 10
[2008/08/11 v1.8] . . . 10
[2016/05/16 v1.9] . . . 10

6 Index **10**

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

1 Usage

The package `pagesel` is a LaTeX 2_ε package:

`\usepackage[`⟨*options*⟩`]{pagesel}`

(For plainTeX and LaTeX 2.09 the similar package `selectp`[1] from Donald Arsenau[2] can be used.)

Depending on the options the package works in two modes:

1. If no page selecting option is present, so the package ignores the other options and finishes itself. So no page will be suppressed by the package and auxiliary files will be written.

2. With at least one page selecting option the specified pages are selected and the other are suppressed. The default for this mode is that auxiliary will not be overwritten. (This can be changed by an option.)

1.1 Page selecting

The package `pagesel` sets up a new counter that is incremented by each `\shipout`. In this way the package counts the output pages regardless the value of the page counter. So each page can individually by addressed, even if there are several pages with the same page number.

1.1.1 Options for selecting pages

odd: The output pages must have an odd number. All even output pages are suppressed. If there are no page areas specified so all odd pages are print. With page areas only the odd pages in this areas are selected.

even: The opposite of option `odd`.

Page area: A page area consists of three elements: the starting output page number, an "area" hyphen, and the output page number of the last page in this area. Each component is optional, so there are four kinds to spezify a page area:

⟨***m***⟩**-**⟨***n***⟩**:** All pages between ⟨*m*⟩ and ⟨*n*⟩ inclusive.

-⟨***n***⟩**:** All pages until ⟨*n*⟩ inclusive.

⟨***m***⟩**-:** The page area starts with ⟨*m*⟩ and all pages to the end of document are selected.

-: All pages (not very useful).

⟨***s***⟩**:** The single page ⟨*s*⟩.

1.1.2 Examples

Options	Output pages
`[1, 4, 9]`	1, 4, and 9
`[7-10, 3]`	3, 7, 8, 9, and 10
`[odd, 3-6]`	3, and 5
`[-4, 3, even, 7-8]`	2, 4, and 8

1.2 Auxiliary files

If a page is suppressed, the `\write` commands are not performed. Labels, index entries, or entries for the table of contents aren't written. So it is likely that the table of contents, registers, and lists are incomplete.

[1] Url: http://ctan.org/pkg/selectp

[2] Donald Arsenau's email address: asnd@triumf.ca

1.2.1 Options for handling auxiliary files

nofiles: This is the default. Auxiliary files are read but not written or changed. Also the job is aborted after the last selected page for saving time.

nonofiles/files: Auxiliary files are written.

1.2.2 Package hyperref

In old versions of `hyperref` [1999/04/12 v6.55] (and below) there is a bug with `\nofiles`:

- Some "garbage" appears on terminal and in the log file. This is harmless and can be ignored.
- The outline auxiliary file `\jobname.out`, however, is opened and truncated to zero bytes. Version 1.0 of this package had loaded a patch file `hypnofil.tex`, if it detects `hyperref` to get `\nofiles` work.

  With the new version of `hyperref` [1999/04/13 v6.56] `\nofiles` works now. Therefore the workaround code is no longer needed and removed.

2 Implementation

2.1 Package

```
⟨*package⟩
\NeedsTeXFormat{LaTeX2e}
\ProvidesPackage{pagesel}
  [2016/05/16 v1.9 Select pages of a document for output (HO)]%
```

If the package is loaded twice, the package code does not work. So stop loading the package, if it is already loaded.

```
\@ifundefined{ps@makevoid}{}{%
  \PackageWarningNoLine{pagesel}{Package already loaded.}%
  \endinput
}
```

\ps@makevoid Macro `\ps@makevoid` clears the output box. Because nothing is shipped out and this is intended, we reduce the counter `\deadcycles` in order to avoid problems, if more than `\maxdeadcycles` pages are omitted.

```
\newcommand*{\ps@makevoid}{%
  \global\setbox\@cclv\copy\voidb@x
  \begingroup
    \count@=\deadcycles
    \advance\count@ by -1\relax
    \deadcycles=\count@
  \endgroup
}
```

\ps@oddpages

```
\newcommand*\ps@oddpages{0}
\DeclareOption{odd}{\renewcommand*\ps@oddpages{1}}
\DeclareOption{even}{\renewcommand*\ps@oddpages{2}}

\DeclareOption{nofiles}{\let\ps@nofiles\nofiles}
\DeclareOption{nonofiles}{\let\ps@nofiles\@empty}
\DeclareOption{files}{\let\ps@nofiles\@empty}
\ExecuteOptions{nofiles}

\DeclareOption*{%
  \begingroup
    \expandafter\ps@checkoption\CurrentOption-\END
    \edef\x{\endgroup\noexpand\ps@store{\ps@first}{\ps@last}}%
```

```
  \x
}
```

\ps@checkoption

```
\newcommand\ps@checkoption{}
\def\ps@checkoption#1-#2\END{%
  \ifx\\#2\\%
    \ifx\\#1\\%
      % empty option
      \def\ps@first{\maxdimen}%
      \def\ps@last{\maxdimen}%
    \else
      \edef\ps@first{#1}%
      \edef\ps@last{#1}%
    \fi
  \else
    \ifx\\#1\\%
      \def\ps@first{-\maxdimen}%
    \else
      \edef\ps@first{#1}%
    \fi
    \ps@checklast#2%
  \fi
}
```

\ps@checklast

```
\newcommand\ps@checklast{}
\def\ps@checklast#1-{%
  \ifx\\#1\\%
    \def\ps@last{\maxdimen}%
  \else
    \edef\ps@last{#1}%
  \fi
}
```

\ps@store

```
\newcommand*{\ps@store}[2]{%
  \expandafter\def\expandafter\ps@testlist\expandafter{%
    \ps@testlist\ps@pagetest{#1}{#2}%
  }%
}
```

\ps@testlist

```
\newcommand*\ps@testlist{}

\ProcessOptions

\begingroup
  \edef\x{%
    \ifnum\ps@oddpages>0 \relax\fi
    \ifx\ps@testlist\@empty\else\relax\fi
  }%
  \ifx\x\@empty
    \endgroup
    \PackageInfo{pagesel}{Nothing to do}%
    \expandafter\endinput
  \fi
\endgroup

\RequirePackage{everyshi}

\ps@nofiles
```

\c@ps@count

```
\newcounter{ps@count}
\setcounter{ps@count}{0}
```

\ps@ReturnAfterElseFi
\ps@ReturnAfterFi

```
\long\def\ps@ReturnAfterElseFi#1\else#2\fi{\fi#1}
\long\def\ps@ReturnAfterFi#1\fi{\fi#1}

\newcommand{\ps@lastpage}{\maxdimen}
\ifx\ps@nofiles\nofiles
  \ifx\ps@testlist\@empty
  \else
    \def\ps@lastpage{0}%
    \newcommand*{\ps@pagetest}[2]{%
      \ifnum#2>\ps@lastpage\relax
        \def\ps@lastpage{#2}%
      \fi
    }%
    \ps@testlist
    \let\ps@pagetest\relax
  \fi
\fi
```

\ps@ifinset

```
\newcommand*{\ps@ifinset}[4]{%
  \ifnum#1>\value{ps@count}%
    \ps@ReturnAfterElseFi{#4}%
  \else
    \ps@ReturnAfterFi{%
      \ifnum#2<\value{ps@count}%
        \ps@ReturnAfterElseFi{#4}%
      \else
        \ps@ReturnAfterFi{#3}%
      \fi
    }%
  \fi
}
```

\ps@pagetest

```
\newcommand*{\ps@pagetest}[2]{%
  \ps@ifinset{#1}{#2}{\let\ps@next\@empty}{}%
}

\EveryShipout{%
  \stepcounter{ps@count}%
  \ifnum\value{ps@count}>\ps@lastpage\relax
    \global\output{%
      \ps@cleanup@if
      \ps@group@message
      \typeout{%
        Package pagesel Notice: Aborting LaTeX job %
        after last selected page (\ps@lastpage).%
      }%
      \ps@message@ignore
      \global\setbox\@cclv\box\voidb@x
      \deadcycles0\relax
```

First leave the output group before ending the job.

```
      \aftergroup\@@end
    }%
  \fi
  \let\ps@next\@empty
  \ifx\ps@testlist\@empty
```

```
  \else
    \let\ps@next\ps@makevoid
    \ps@testlist
  \fi
  \ifnum\ps@oddpages=1 %
    \ifodd\value{ps@count}%
    \else
      \let\ps@next\ps@makevoid
    \fi
  \fi
  \ifnum\ps@oddpages=2 %
    \ifodd\value{ps@count}%
      \let\ps@next\ps@makevoid
    \else
    \fi
  \fi
  \ps@begindvi
  \ps@next
}
\begingroup\expandafter\expandafter\expandafter\endgroup
\expandafter\ifx\csname currentiflevel\endcsname\relax
  \let\ps@cleanup@if\@empty
\else
  \def\ps@cleanup@if{%
    \ifnum\currentiflevel>\@ne
      \csname fi\endcsname
      \expandafter\ps@cleanup@if
    \fi
  }%
\fi
```

Because of `\aftergroup` it is too dangerous to perform a similar cleanup for groups.

```
\begingroup\expandafter\expandafter\expandafter\endgroup
\expandafter\ifx\csname currentgrouplevel\endcsname\relax
  \let\ps@group@message\@empty
  \def\ps@message@ignore{%
    \typeout{%
      (pagesel) \space\space\@spaces\@spaces\@spaces
      Messages (\string\end\space occurred ...) can be ignored.%
    }%
  }%
\else
  \def\ps@group@message{%
    \ifnum\currentgrouplevel>\@ne
      \def\ps@message@ignore{%
        \typeout{%
          (pagesel) \space\space\@spaces\@spaces\@spaces
          Message (\string\end\space occurred ...) %
          can be ignored.%
        }%
      }%
    \else
      \let\ps@message@ignore\@empty
    \fi
  }%
\fi
```

2.2 AtBeginDvi hook support

The material of box `\@begindvibox` is recorded in parallel in box `\ps@begindvibox`.

```
\newbox\ps@begindvibox
```

```
\ifvoid\@begindvibox
\else
  \global\setbox\ps@begindvibox\vbox{%
    \unvbox\@begindvibox
  }%
\fi
\let\ps@org@AtBeginDvi\AtBeginDvi
\def\AtBeginDvi#1{%
  \global\setbox\ps@begindvibox\vbox{%
    \unvbox\ps@begindvibox
    #1%
  }%
  \ps@org@AtBeginDvi{#1}%
}
```

\ps@begindvi Macro \ps@begindvi is called the similar way as \@begindvi. If the first page is printed, then \AtBeginDvi should work as usual. Otherwise the contents of box \ps@begindvibox is set on the first selected page.

```
\def\ps@begindvi{%
  \ifx\ps@next\@empty
    \global\let\ps@begindvi\@empty
  \else
    \global\let\ps@begindvi\ps@begindvi@do
  \fi
}
\def\ps@begindvi@do{%
  \ifx\ps@next\@empty
    \setbox\@cclv\vbox{%
      \unvbox\ps@begindvibox
      \box\@cclv
    }%
    \global\let\ps@begindvi\@empty
  \fi
}
```

```
⟨/package⟩
```

3 Installation

3.1 Download

Package. This package is available on CTAN[3]:

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

CTAN:macros/latex/contrib/oberdiek/pagesel.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.

3.2 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
```

[3] http://ctan.org/pkg/pagesel

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/
```

3.3 Package installation

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

```
tex pagesel.dtx
```

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

```
pagesel.sty  → tex/latex/oberdiek/pagesel.sty
pagesel.pdf → doc/latex/oberdiek/pagesel.pdf
pagesel.dtx → source/latex/oberdiek/pagesel.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.

3.4 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.

3.5 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{pagesel.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 pagesel.dtx
makeindex -s gind.ist pagesel.idx
pdflatex pagesel.dtx
makeindex -s gind.ist pagesel.idx
pdflatex pagesel.dtx
```

4 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 pagesel.xml.

```
⟨*catalogue⟩
<?xml version='1.0' encoding='us-ascii'?>
<!DOCTYPE entry SYSTEM 'catalogue.dtd'>
<entry datestamp='$Date$' modifier='$Author$' id='pagesel'>
  <name>pagesel</name>
  <caption>Select pages of a document for output.</caption>
  <authorref id='auth:oberdiek'/>
  <copyright owner='Heiko Oberdiek' year='1999,2003,2006-2008'/>
  <license type='lppl1.3'/>
  <version number='1.9'/>
  <description>
    Selects single pages, ranges of pages, odd pages or even pages
    for output.
    <p/>
    The package is part of the <xref refid='oberdiek'>oberdiek</xref>
    bundle.
  </description>
  <documentation details='Package documentation'
      href='ctan:/macros/latex/contrib/oberdiek/pagesel.pdf'/>
  <ctan file='true' path='/macros/latex/contrib/oberdiek/pagesel.dtx'/>
  <miktex location='oberdiek'/>
  <texlive location='oberdiek'/>
  <install path='/macros/latex/contrib/oberdiek/oberdiek.tds.zip'/>
</entry>
⟨/catalogue⟩
```

5 History

[1999/03/01 v0.9]

- The first version was built as a response to a question of Dirk Kuypers[4], published in the newsgroup de.comp.text.tex:
  "Re: pdflatex nur fuer bestimmte Seiten?!?"[5]

[1999/04/05 v1.0]

- Documentation added in dtx format.
- Copyright: LPPL (CTAN:macros/latex/base/lppl.txt)
- Options odd, even added.
- \nofiles added, bug fix for hyperref.
- Abort loading of package, if nothing to do.

[1999/04/13 v1.1]

- \nofiles bug fix removed because of hyperref 6.55.
- First CTAN release.

[4] Dirk Kuypers's email address: dk@comnets.rwth-aachen.de
[5] Url: http://groups.google.com/group/de.comp.text.tex/msg/6b68c7b3439fb658

[2003/06/05 v1.2]

- \deadcyles is decremented for omitted pages.
- LPPL 1.2.

[2006/02/20 v1.3]

- Code is not changed.
- New DTX framework.
- LPPL 1.3

[2006/03/02 v1.4]

- Support for \AtBeginDvi added.

[2006/03/07 v1.5]

- Job is aborted after last selected page.

[2007/04/11 v1.6]

- Line ends sanitized.

[2007/04/12 v1.7]

- Hard coded box number 255 replaced by macro \@cclv.

[2008/08/11 v1.8]

- Code is not changed.
- URL updated from www.dejanews.com to groups.google.com.

[2016/05/16 v1.9]

- Documentation updates.

6 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

\@@end 125
\@begindvibox 185, 188
\@cclv 10, 123, 208, 210
\@empty 21, 22, 68, 70, 84, 110, 128, 129, 151, 162, 180, 200, 201, 207, 212
\@ifundefined 5
\@ne 154, 171
\@spaces 165, 174
\\ 32, 33, 42, 52

A

\advance 13
\aftergroup 125
\AtBeginDvi 191, 192

B

\box 123, 210

C

\c@ps@count 78
\copy 10
\count@ 12, 13, 14
\csname 150, 155, 161
\currentgrouplevel 171
\currentiflevel 154
\CurrentOption 26

D

\deadcycles 12, 14, 124

\DeclareOption . . 18, 19, 20, 21, 22, 24

E

\END 26, 31
\end 166, 175
\endcsname 150, 155, 161
\endinput 7, 73
\EveryShipout 112
\ExecuteOptions 23

I

\ifnum 67, 88, 97, 101, 114, 134, 140, 154, 171
\ifodd 135, 141
\ifvoid 185
\ifx 32, 33, 42, 52, 68, 70, 83, 84, 129, 150, 161, 200, 207

M

\maxdimen 35, 36, 43, 53, 82

N

\NeedsTeXFormat 2
\newbox 184
\newcommand 9, 17, 30, 50, 58, 63, 82, 87, 96, 109
\newcounter 78
\nofiles 20, 83

O

\output 115

P

\PackageInfo 72
\PackageWarningNoLine 6
\ProcessOptions 64
\ProvidesPackage 3
\ps@begindvi 146, 199
\ps@begindvi@do 203, 206
\ps@begindvibox 184, 187, 193, 194, 209
\ps@checklast 47, 50
\ps@checkoption 26, 30
\ps@cleanup@if 116, 151, 153, 156
\ps@first 27, 35, 38, 43, 45
\ps@group@message 117, 162, 170
\ps@ifinset 96, 110
\ps@last 27, 36, 39, 53, 55
\ps@lastpage . . . 82, 86, 88, 89, 114, 120
\ps@makevoid 9, 131, 137, 142
\ps@message@ignore 122, 163, 172, 180
\ps@next 110, 128, 131, 137, 142, 147, 200, 207
\ps@nofiles 20, 21, 22, 77, 83
\ps@oddpages 17, 67, 134, 140
\ps@org@AtBeginDvi 191, 197
\ps@pagetest 60, 87, 93, 109
\ps@ReturnAfterElseFi 80, 98, 102
\ps@ReturnAfterFi 80, 100, 104
\ps@store 27, 58
\ps@testlist 59, 60, 63, 68, 84, 92, 129, 132

R

\renewcommand 18, 19
\RequirePackage 76

S

\setbox 10, 123, 187, 193, 208
\setcounter 79
\space 165, 166, 174, 175
\stepcounter 113

T

\typeout 118, 164, 173

U

\unvbox 188, 194, 209

V

\value 97, 101, 114, 135, 141
\vbox 187, 193, 208
\voidb@x 10, 123

X

\x 27, 28, 66, 70