The pdflscape package

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

2016/05/14 v0.11

Abstract

Package pdflscape adds PDF support to the environment `landscape` of package lscape by setting the PDF page attribute `/Rotate`.

Contents

1 Documentation **2**
- 1.1 Supported drivers . . . 2
- 1.2 Caveat . . . 2
- 1.3 Requirements . . . 2
- 1.4 Usage . . . 2

2 Implementation **2**
- 2.1 Package identification . . . 2
- 2.2 Driver options . . . 3
- 2.3 Autodetection of driver . . . 3
  - 2.3.1 Driver pdftex . . . 3
  - 2.3.2 Driver xetex . . . 4
  - 2.3.3 Detect driver based on `\Gin@driver` . . . 4
  - 2.3.4 Driver dvips . . . 4
- 2.4 Driver implementation . . . 5
  - 2.4.1 pdfTeX . . . 5
  - 2.4.2 PostScript driver . . . 6
  - 2.4.3 Driver dvipdfm . . . 6
- 2.5 Driver independent stuff . . . 6

3 Test **7**
- 3.1 Driver detection tests . . . 7
- 3.2 Test for `\PLS@CheckAngle` . . . 8
- 3.3 Test for rotate . . . 9

4 Installation **11**
- 4.1 Download . . . 11
- 4.2 Bundle installation . . . 11
- 4.3 Package installation . . . 11
- 4.4 Refresh file name databases . . . 12
- 4.5 Some details for the interested . . . 12

5 Catalogue **12**

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

6 History **13**
[2001/01/15 v0.1] . . . 13
[2001/02/04 v0.2] . . . 13
[2004/05/11 v0.3] . . . 13
[2004/05/12 v0.4] . . . 13
[2006/02/20 v0.5] . . . 13
[2006/04/24 v0.6] . . . 13
[2007/04/11 v0.7] . . . 13
[2007/04/17 v0.8] . . . 14
[2007/10/21 v0.9] . . . 14
[2008/08/11 v0.10] . . . 14
[2016/05/14 v0.11] . . . 14

7 Index **14**

1 Documentation

1.1 Supported drivers

- pdftex,luatex
- dvips, dvipsone, pctex32, pctexps (and other drivers that provide a non-empty `\Gin@PS@raw`)
- dvipdfm

1.2 Caveat

Depending on the configuration Ghostscript adds a guessed rotation entry by its own. This can lead to two `/Rotate` entries per page. To prevent this behaviour set the parameter `AutoRotatePages` to `/None`, eg:

```
ps2pdf -dAutoRotatePages=/None
```

1.3 Requirements

- The package lscape.
- The package atbegshi for all drivers except pdftex,luatex.

1.4 Usage

Load this package instead of or after package lscape:

```
\usepackage{pdflscape}
\begin{landscape}...\end{landscape}
```

2 Implementation

2.1 Package identification

1 ⟨*package⟩

Package identification.

```
2 \NeedsTeXFormat{LaTeX2e}
3 \ProvidesPackage{pdflscape}%
```

```
  [2016/05/14 v0.11 Display of landscape pages in PDF (HO)]
```

2.2 Driver options

```
\let\PLS@option\@empty
\let\PLS@driver\@empty
\def\PLS@temp#1#2{%
  \DeclareOption{#1}{%
    \let\PLS@option\CurrentOption
    \def\PLS@driver{#2}%
  }%
}
\PLS@temp{pdftex}{pdftex}
\PLS@temp{luatex}{pdftex}
\PLS@temp{dvips}{dvips}
\PLS@temp{dvipsone}{dvips}
\PLS@temp{pctexps}{dvips}
\PLS@temp{dvipdfm}{dvipdfm}
\PLS@temp{dvipdfmx}{dvipdfm}
\PLS@temp{xetex}{dvipdfm}
\let\PLS@temp\@empty
\DeclareOption*{%
  \ifx\PLS@temp\@empty
    \let\PLS@temp\CurrentOption
  \else
    \edef\PLS@temp{\PLS@temp,\CurrentOption}%
  \fi
}
\ProcessOptions*
\ifx\PLS@option\@empty
\else
  \expandafter\PassOptionsToPackage\expandafter{\PLS@option}{lscape}%
\fi
\ifx\PLS@temp\@empty
\else
  \expandafter\PassOptionsToPackage\expandafter{\PLS@temp}{lscape}%
\fi
\RequirePackage{lscape}
```

2.3 Autodetection of driver

2.3.1 Driver **pdftex**

```
\RequirePackage{ifpdf}[2006/02/20]
\def\PLS@temp{pdftex}
\ifpdf
  \ifx\PLS@driver\@empty
    \let\PLS@driver\PLS@temp
    \PackageInfo{pdflscape}{Auto-detected driver: \PLS@temp}%
  \fi
  \ifx\PLS@driver\PLS@temp
  \else
    \PackageError{pdflscape}{%
      The driver option `\PLS@option' is not compatible\MessageBreak
      with pdfTeX in PDF mode, using driver `pdftex' instead%
    }\@ehc
    \let\PLS@driver\PLS@temp
  \fi
\fi
```

2.3.2 Driver xetex

```
\RequirePackage{ifxetex}%
\def\PLS@temp{dvipdfm}
\ifxetex
  \ifx\PLS@driver\@empty
    \let\PLS@driver\PLS@temp
    \PackageInfo{pdflscape}{%
      Auto-detected driver: \PLS@temp\space(xetex)%
    }%
  \fi
  \ifx\PLS@driver\PLS@temp
  \else
    \PackageError{pdflscape}{%
      The driver option `\PLS@option' is not compatible\MessageBreak
      with XeTeX, using driver for XeTeX (dvipdfm) instead%
    }\@ehc
    \let\PLS@driver\PLS@temp
  \fi
\fi
```

2.3.3 Detect driver based on \Gin@driver

```
\def\PLS@temp#1#2{%
  \ifx\PLS@driver\@empty
    \begingroup
      \def\PLS@temp{#1.def}%
    \expandafter\endgroup
    \ifx\PLS@temp\Gin@driver
      \def\PLS@driver{#2}%
      \PackageInfo{pdflscape}{%
        Auto-detected driver: \PLS@driver\space(#1)%
      }%
    \fi
  \fi
}
\PLS@temp{dvipdfm}{dvipdfm}
\PLS@temp{dvipdfmx}{dvipdfm}
\PLS@temp{dvips}{dvips}
\PLS@temp{dvipsone}{dvips}
\PLS@temp{pctexps}{dvips}
```

2.3.4 Driver dvips

```
\ifx\PLS@driver\@empty
  \def\PLS@temp{dvips}%
  \ifnum 1=0\ifx\Gin@PS@raw\@undefined
            \else
              \ifx\Gin@PS@raw\relax
              \else
                \ifx\Gin@PS@raw\@empty
                \else
                  1%
                \fi
              \fi
            \fi
            \space
    \let\PLS@driver\PLS@temp
    \PackageInfo{pdflscape}{%
      Auto-detected driver: \PLS@temp\space(PostScript)%
    }%
```

```
  \fi
\fi
\ifx\PLS@driver\@empty
  \PackageError{pdflscape}{%
    Neither pdfTeX, PostScript, nor dvipdfm driver found%
  }\@ehc
\fi
```

2.4 Driver implementation

- `\PLS@AddRotate #1`
  it expects the correct rotation number in `#1` and implements the adding of the /Rotation entry in the `/Page` object of the current page.

- `\PLS@RemoveRotate`
  it removes a previous `/Rotate` entry, if necessary.

2.4.1 pdfTeX

Not too nice is the global setting of `\pdfpageattr`. Perhaps this can be changed in future versions.

luatex compat.

```
\ifx\pdfpageattr\@undefined
  \ifx\pdfvariable\@undefined\else
    \protected\edef\pdfpageattr{\pdfvariable pageattr}
  \fi
\fi
\def\PLS@temp{pdftex}
\ifx\PLS@temp\PLS@driver
  \def\PLS@AddRotate#1{%
    \ifnum#1=0 %
```

Already the default.

```
    \else
      \global\pdfpageattr\expandafter{%
        \the\pdfpageattr
        /Rotate #1%
      }%
    \fi
  }%
```

Removes a `/Rotate` entry.

```
  \def\PLS@RemoveRotate{%
    \begingroup
      \global\pdfpageattr\expandafter{\expandafter}%
      \expandafter\PLS@@RemoveRotate
        \the\pdfpageattr /Rotate\@nil
    \endgroup
  }%
  \def\PLS@@RemoveRotate#1/Rotate#2\@nil{%
```

Append `/Rotate` free stuff to `\pdfpageattr`.

```
    \global\pdfpageattr\expandafter{\the\pdfpageattr#1}%
    \ifx\\#2\\%
```

Ready, because the detected `/Rotate` is part of the end marker: `/Rotate\@nil`

```
    \else
```

First read in the argument of /Rotate, then continue parsing.

```
      \afterassignment\PLS@@RemoveRotate
      \count0=#2\@nil
    \fi
  }%
\fi
```

2.4.2 PostScript driver

```
\def\PLS@temp{dvips}
\ifx\PLS@temp\PLS@driver
  \RequirePackage{atbegshi}%
  \AtBeginShipout{\PLS@AtBeginShipout}%
  \let\PLS@AtBeginShipout\@empty
  \def\PLS@AddRotate#1{%
    \ifnum#1=0 %
    \else
      \def\PLS@AtBeginShipout{%
        \global\setbox\AtBeginShipoutBox\vbox{%
          \Gin@PS@raw{%
            [{ThisPage}\string<\string</Rotate #1\string>\string>%
            /PUT pdfmark%
          }%
          \box\AtBeginShipoutBox
        }%
      }%
    \fi
  }%
  \def\PLS@RemoveRotate{\let\PLS@AtBeginShipout\@empty}%
\fi
```

2.4.3 Driver dvipdfm

```
\def\PLS@temp{dvipdfm}
\ifx\PLS@temp\PLS@driver
  \RequirePackage{atbegshi}%
  \AtBeginShipout{\PLS@AtBeginShipout}%
  \let\PLS@AtBeginShipout\@empty
  \def\PLS@AddRotate#1{%
    \ifnum#1=0 %
    \else
      \def\PLS@AtBeginShipout{%
        \global\setbox\AtBeginShipoutBox\vbox{%
          \special{pdf: put @thispage <</Rotate #1>>}%
          \box\AtBeginShipoutBox
        }%
      }%
    \fi
  }%
  \def\PLS@RemoveRotate{\let\PLS@AtBeginShipout\@empty}%
\fi
```

2.5 Driver independent stuff

The landscape environment is extended by adding the correct /Rotate entries.

```
\g@addto@macro{\landscape}{\PLS@Rotate{90}}
\g@addto@macro{\endlandscape}{\PLS@Rotate{0}}
```

\PLS@Rotate Main macro, that sets the /Rotate entry.

Argument: any TeX number or nothing, that means zero.
Driver independent.

```
\def\PLS@Rotate#1{%
  \begingroup
```

1. Check and validate the argument.

```
    \PLS@CheckAngle{#1}%
  \expandafter\endgroup
```

2. Remove previous `\Rotate` entry.

```
  \expandafter\PLS@RemoveRotate
```

3. Add `/Rotate` entry.

```
  \expandafter\PLS@AddRotate\expandafter{\the\count@}%
}
```

`\PLS@CheckAngle` Validates the rotation angle.
The result is stored in the count register `\count@`.
Driver independent.

```
\def\PLS@CheckAngle#1{%
```

Nothing means zero:

```
  \ifx\\#1\\%
    \count@=0 %
  \else
    \count@=#1\relax
  \fi
```

Normalize to interval -360 < `\count@` < 360.

```
  \@whilenum\count@>359\do{\advance\count@ -360 }%
  \@whilenum\count@<-359\do{\advance\count@ 360 }%
```

Check values: 0, +/-90, +/-180, +/-270

```
  \ifnum 1=0\ifnum\count@=0 \else
             \ifnum\count@=90 \else
             \ifnum\count@=-90 \else
             \ifnum\count@=180 \else
             \ifnum\count@=-180 \else
             \ifnum\count@=270 \else
             \ifnum\count@=-270 \else
            1\fi\fi\fi\fi\fi\fi\fi\relax
    \PackageError{pdflscape}{%
      Invalid value \the\count@\space for /Rotate%
    }\@ehc
    \count@=0 %
  \fi
}
```

⟨/package⟩

3 Test

3.1 Driver detection tests

⟨*test0⟩

```
\NeedsTeXFormat{LaTeX2e}
\documentclass{minimal}
\usepackage{qstest}
\IncludeTests{*}
```

```
\LogTests{log}{*}{*}
⟨/test0⟩
⟨*test1⟩
\usepackage{ifpdf}
\ifpdf
  \def\ExpectDriver{pdftex}%
\fi
\usepackage{ifxetex}%
\ifxetex
  \def\ExpectDriver{dvipdfm}%
\fi
\usepackage{pdflscape}[2016/05/14]
⟨/test1⟩
⟨*test2⟩
\def\ExpectDriver{pdftex}
\usepackage[pdftex]{pdflscape}[2016/05/14]
⟨/test2⟩
⟨*test3⟩
\def\ExpectDriver{dvipdfm}
\usepackage[dvipdfmx]{pdflscape}[2016/05/14]
⟨/test3⟩
⟨*test4⟩
\def\ExpectDriver{dvips}
\usepackage[dvipsone]{graphics}
\usepackage{pdflscape}[2016/05/14]
⟨/test4⟩
⟨*test0⟩
\begin{document}
\begin{qstest}{driver}{driver}
  \makeatletter
  \Expect*{\PLS@driver}*{\ExpectDriver}%
\end{qstest}
\end{document}
⟨/test0⟩
```

3.2 Test for \PLS@CheckAngle

```
⟨*test5⟩
\NeedsTeXFormat{LaTeX2e}
\documentclass{minimal}
\usepackage{qstest}
\IncludeTests{*}
\LogTests{log}{*}{*}
\usepackage{pdflscape}[2016/05/14]
\begin{document}
\begin{qstest}{checkangle}{checkangle}
  \makeatletter
  \def\Result{%
    \def\ResultString{\the\count@}%
  }%
  \def\PackageError#1#2#3{%
    \def\Result{%
      \def\ResultString{error}%
    }%
  }%
  \def\Test#1#2{%
    \begingroup
```

```
      \PLS@CheckAngle{#1}%
      \Result
      \Expect*{\ResultString}{#2}%
    \endgroup
  }%
  \Test{0}{0}%
  \Test{90}{90}%
  \Test{180}{180}%
  \Test{270}{270}%
  \Test{360}{0}%
  \Test{450}{90}%
  \Test{540}{180}%
  \Test{630}{270}%
  \Test{720}{0}%
  \Test{3600}{0}%
  \Test{3690}{90}%
  \Test{-90}{270}%
  \Test{-180}{180}%
  \Test{-270}{-90}%
  \Test{-360}{0}%
  \Test{-450}{270}%
  \Test{540}{180}%
  \Test{-630}{90}%
  \Test{-720}{0}%
  \Test{-3600}{0}%
  \Test{-3690}{270}%
  \Test{1}{error}%
  \Test{-1}{error}%
  \Test{123}{error}%
\end{qstest}
\end{document}
```
⟨/test5⟩

3.3 Test for rotate

⟨*test6⟩
```
\NeedsTeXFormat{LaTeX2e}
\documentclass{article}
\usepackage{pdflscape}[2016/05/14]
\begin{document}
\section{Portrait page}
\begin{landscape}
\section{Landscape page}
Foo bar foo bar foo bar foo bar foo bar
foo bar foo bar foo bar foo bar foo bar.
\end{landscape}
\section{Portrait Section}
Foo bar
\begin{landscape}
\section{Again Landscape Section}
\newpage
\section{Second Page of Landscape Section}
\end{landscape}
\section{Second Last Portrait Page}
\newpage
\section{Last Portrait Page}
\end{document}
```
⟨/test6⟩

⟨*result6⟩

```
/Page 1 /Rotate 0
/Page 2 /Rotate 90
/Page 3 /Rotate 0
/Page 4 /Rotate 90
/Page 5 /Rotate 90
/Page 6 /Rotate 0
/Page 7 /Rotate 0
```
⟨/result6⟩

⟨*ExtractRotate⟩
```
/**
 * ExtractRotate.java
 *
 * Copyright (C) 2007 by Heiko Oberdiek <heiko.oberdiek at googlemail.com>
 *
 * Requires: PDFBox (http://www.pdfbox.org/)
 *
 * Syntax: java ExtractRotate <pdffile> <textfile>
 *
 * The <pdffile> is analyzed and for each page its rotation
 * setting is printed in the <textfile>. Example:
 *   /Page 1 /Rotate 0
 *   /Page 2 /Rotate 90
 */
import java.io.FileInputStream;
import java.io.FileWriter;
import org.pdfbox.pdfparser.PDFParser;
import org.pdfbox.pdmodel.PDDocument;
import org.pdfbox.pdmodel.PDDocumentCatalog;
import org.pdfbox.pdmodel.PDPage;

public class ExtractRotate {

    public static void main(String[] args) {
        try {
            String infile = args[0];
            String outfile = args[1];
            FileWriter out = new FileWriter(outfile);
            PDFParser parser =
                    new PDFParser(new FileInputStream(infile));
            parser.parse();
            PDDocument document = parser.getPDDocument();
            PDDocumentCatalog catalog = document.getDocumentCatalog();
            int i = 0;
            for (Object page: catalog.getAllPages()) {
                i++;
                out.write("/Page " + i + " " + "/Rotate "
                        + ((PDPage)page).findRotation() + "\n");
            }
            document.close();
            out.close();
        }
        catch (Exception e) {
            e.printStackTrace();
            System.exit(1);
        }
    }
}
```
⟨/ExtractRotate⟩

4 Installation

4.1 Download

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

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

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

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 pdflscape.dtx
```

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

```
pdflscape.sty            → tex/latex/oberdiek/pdflscape.sty
pdflscape.pdf            → doc/latex/oberdiek/pdflscape.pdf
test/pdflscape-test1.tex → doc/latex/oberdiek/test/pdflscape-test1.tex
test/pdflscape-test2.tex → doc/latex/oberdiek/test/pdflscape-test2.tex
test/pdflscape-test3.tex → doc/latex/oberdiek/test/pdflscape-test3.tex
test/pdflscape-test4.tex → doc/latex/oberdiek/test/pdflscape-test4.tex
test/pdflscape-test5.tex → doc/latex/oberdiek/test/pdflscape-test5.tex
test/pdflscape-test6.tex → doc/latex/oberdiek/test/pdflscape-test6.tex
test/pdflscape-test6.txt → doc/latex/oberdiek/test/pdflscape-test6.txt
test/ExtractRotate.java  → doc/latex/oberdiek/test/ExtractRotate.java
pdflscape.dtx            → source/latex/oberdiek/pdflscape.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.

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

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

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 pdflscape.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{pdflscape.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 pdflscape.dtx
makeindex -s gind.ist pdflscape.idx
pdflatex pdflscape.dtx
makeindex -s gind.ist pdflscape.idx
pdflatex pdflscape.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 `pdflscape.xml`.

```
391 ⟨*catalogue⟩
392 <?xml version='1.0' encoding='us-ascii'?>
393 <!DOCTYPE entry SYSTEM 'catalogue.dtd'>
394 <entry datestamp='$Date$' modifier='$Author$' id='pdflscape'>
395   <name>pdflscape</name>
396   <caption>Make landscape pages display as landscape.</caption>
397   <authorref id='auth:oberdiek'/>
398   <copyright owner='Heiko Oberdiek' year='2001,2004,2006-2008'/>
399   <license type='lppl1.3'/>
400   <version number='0.11'/>
401   <description>
402     The package adds PDF support to the landscape environment of
```

```
403     package <xref refid='lscape'>lscape</xref>, by setting the PDF
404     <tt>/Rotate</tt> page attribute.  Pages with this attribute will be
405     displayed in landscape orientation by conforming PDF viewers.
406     <p/>
407     The package is part of the <xref refid='oberdiek'>oberdiek</xref> bundle.
408   </description>
409   <documentation details='Package documentation'
410       href='ctan:/macros/latex/contrib/oberdiek/pdflscape.pdf'/>
411   <ctan file='true' path='/macros/latex/contrib/oberdiek/pdflscape.dtx'/>
412   <miktex location='oberdiek'/>
413   <texlive location='oberdiek'/>
414   <install path='/macros/latex/contrib/oberdiek/oberdiek.tds.zip'/>
415 </entry>
```
416 ⟨/catalogue⟩

6 History

[2001/01/15 v0.1]

- First public version, published in de.comp.text.tex: "Re: Querformat, pdfLaTex und thumbpdf"[2]

[2001/02/04 v0.2]

- Minor documentation update.
- CTAN.

[2004/05/11 v0.3]

- Support for dvipdfm added.

[2004/05/12 v0.4]

- Bug fix: support for multipage landscape environment for all drivers except for pdftex.

[2006/02/20 v0.5]

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

[2006/04/24 v0.6]

- Bug fix: `\PLS@RemoveRotate` reinserted for `\PLS@Rotate`.
- Fix for `\PLS@RemoveRotate` (dvips/dvipdfm): `\PLS@EverhShi` hook is cleared.

[2007/04/11 v0.7]

- Line ends sanitized.

[2]Url: http://groups.google.com/group/de.comp.text.tex/msg/e054c5795e52d2b8

[2007/04/17 v0.8]

- Package atbegshi replaces everyshi.

[2007/10/21 v0.9]

- Driver detection for XeTeX added.
- Fix for rotation angles $\leq$ -360 or $\geq$ 360.

[2008/08/11 v0.10]

- Code is not changed.
- URLs updated.

[2016/05/14 v0.11]

- LuaTeX Compatibility

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

Symbols

\@ehc 51, 69, 113, 213
\@empty . 5, 6, 21, 23, 30, 34, 42, 58, 74, 91, 97, 110, 151, 166, 172, 184
\@nil 135, 138, 143
\@undefined 93, 115, 116
\@whilenum 201, 202
\\ 140, 196

A

\advance 201, 202
\afterassignment 142
\AtBeginShipout 150, 171
\AtBeginShipoutBox . 156, 161, 177, 179

B

\begin 250, 251, 264, 265, 313, 315, 322
\box 161, 179

C

\count 143
\count@ 193, 197, 199, 201, 202, 203, 204, 205, 206, 207, 208, 209, 212, 214, 268
\CurrentOption 9, 24, 26

D

\DeclareOption 8, 22
\do 201, 202
\documentclass 220, 259, 311

E

\end . . 254, 255, 306, 307, 319, 326, 330
\endlandscape 187
\Expect 253, 279
\ExpectDriver 228, 232, 237, 241, 245, 253

G

\g@addto@macro 186, 187
\Gin@driver 78
\Gin@PS@raw 93, 95, 97, 157

I

\ifnum 93, 123, 153, 174, 203, 204, 205, 206, 207, 208, 209
\ifpdf 41, 227
\ifx 23, 30, 34, 42, 46, 58, 64, 74, 78, 91, 93, 95, 97, 110, 115, 116, 121, 140, 148, 169, 196
\ifxetex 57, 231
\IncludeTests 222, 261

L

\landscape 186
\LogTests 223, 262

M

\makeatletter 252, 266
\MessageBreak 49, 67

N

\n 379

\NeedsTeXFormat 2, 219, 258, 310
\newpage 324, 328

P

\PackageError . . . 48, 66, 111, 211, 270
\PackageInfo 44, 60, 80, 105
\PassOptionsToPackage 32, 36
\pdfpageattr 115, 117, 125, 126, 133, 135, 139
\pdfvariable 116, 117
\PLS@@RemoveRotate 134, 138, 142
\PLS@AddRotate 122, 152, 173, 193
\PLS@AtBeginShipout 150, 151, 155, 166, 171, 172, 176, 184
\PLS@CheckAngle 190, 195, 277
\PLS@driver 6, 10, 42, 43, 46, 52, 58, 59, 64, 70, 74, 79, 81, 91, 104, 110, 121, 148, 169, 253
\PLS@option 5, 9, 30, 32, 49, 67
\PLS@RemoveRotate . 131, 166, 184, 192
\PLS@Rotate 186, 187, 188
\PLS@temp 7, 13, 14, 15, 16, 17, 18, 19, 20, 21, 23, 24, 26, 34, 36, 40, 43, 44, 46, 52, 56, 59, 61, 64, 70, 73, 76, 78, 86, 87, 88, 89, 90, 92, 104, 106, 120, 121, 147, 148, 168, 169
\ProcessOptions 29
\protected 117
\ProvidesPackage 3

R

\RequirePackage . . 38, 39, 55, 149, 170
\Result 267, 271, 278
\ResultString 268, 272, 279

S

\section 314, 316, 320, 323, 325, 327, 329
\setbox 156, 177
\space 61, 81, 103, 106, 212
\special 178

T

\Test . . 275, 282, 283, 284, 285, 286, 287, 288, 289, 290, 291, 292, 293, 294, 295, 296, 297, 298, 299, 300, 301, 302, 303, 304, 305
\the 126, 135, 139, 193, 212, 268

U

\usepackage 221, 226, 230, 234, 238, 242, 246, 247, 260, 263, 312

V

\vbox 156, 177