The printfilelist package

Gorkem Aksaray
aksarayg@tcd.ie

v1.1.0 from 2026/06/07

Source repo: https://github.com/gaksaray/printfilelist.

1 Introduction

This is a LaTeX package for inspecting the files used by the current document. It provides two commands: `\printfilelist`, which prints a list of all the files (with an option to filter them), and `\ispackageloaded`, which prints whether a given package is loaded or not. Both commands can be issued at any point within the document body. The package has no dependencies beyond the LaTeX kernel.

2 Usage

Load the package anywhere in the preamble:

```
\usepackage{printfilelist}
```

The load position does not matter: the package records every input file used by the document, and the two commands below read that list wherever they are called.

`\printfilelist[`⟨*filter*⟩`]` Prints the files used by the current document, one per line, together with their version information. The optional ⟨*filter*⟩ keeps only the files whose name contains the given text; for example,

```
\printfilelist[.sty]
```

lists only the `.sty` files. The default filter, `.`, keeps any file that has an extension, while an empty filter, `\printfilelist[]`, keeps every recorded file. If nothing matches, a short note is printed instead.

`\printfilelist*[`⟨*filter*⟩`]` The starred form prints a compact, comma-separated list of file names only, without version information. It is set as ordinary inline text, so it may be embedded in running prose, as in `Files: \printfilelist*[]`.

`\ispackageloaded{`⟨*package*⟩`}` Prints whether ⟨*package*⟩ is loaded, together with its version if it is. The starred form, `\ispackageloaded*`, prints only "is loaded" or "is not loaded". This command recognizes packages (`.sty` files) only, not document classes.

3 Notes

The package issues `\listfiles` when loaded; this is what makes the kernel keep the file list until the commands above are used. As a side effect, LaTeX also writes the file list to the `.log` at the end of the run. The package always omits its own entry, `printfilelist.sty`. The detailed form of `\printfilelist` forms its own paragraph, while the starred form is inline; neither is intended for a moving argument such as a section title.

4 Implementation

The kernel accumulates the names of all input files in `\@filelist`, but discards it at `\begin{document}` unless `\listfiles` has been issued. Issuing it here keeps the list available for the commands below.

```
\listfiles
```

`\ispackageloaded` The kernel records a loaded package in the control sequence `\ver@`⟨*name*⟩`.sty`. The unstarred form prints that version; the starred form reports only whether the package is present.

```
\newcommand{\ispackageloaded}{\@ifstar\pfl@ispl@star\pfl@ispl@plain}
\newcommand{\pfl@ispl@plain}[1]{%
    \if\relax\detokenize{#1}\relax\else
        \ifcsname ver@#1.sty\endcsname
            \texttt{#1.sty}: \csname ver@#1.sty\endcsname%
        \else
            \texttt{#1} is not loaded%
        \fi
    \fi
}
\newcommand{\pfl@ispl@star}[1]{%
    \if\relax\detokenize{#1}\relax\else
        \ifcsname ver@#1.sty\endcsname
            \texttt{#1} is loaded%
        \else
            \texttt{#1} is not loaded%
        \fi
    \fi
}
```

`\printfilelist` The star selects the compact, inline form; without it, the detailed one-per-line listing is produced. Either way the list is read live from `\@filelist` when the command is called.

```
\newif\ifpfl@inline
\newif\ifpfl@found
\newcommand{\printfilelist}{\@ifstar{\pfl@inlinetrue\pfl@run}{\pfl@inlinefalse\pfl@run}}
\newcommand{\pfl@run}[1][.]{%
    \begingroup
    \ifpfl@inline\else\par\fi
    \pfl@foundfalse
```

```
    \@for\pfl@file:=\@filelist\do{%
        \expandafter\pfl@test\expandafter{\pfl@file}{#1}%
    }%
    \ifpfl@found\else
        \ifpfl@inline\pfl@nomatch\else\noindent\pfl@nomatch\par\fi
    \fi
    \ifpfl@inline\else\par\fi
    \endgroup
}
```

The message shown when nothing matches the filter; redefine it to change the wording. Its framing (inline or paragraph) is supplied by `\pfl@run`.

```
\newcommand{\pfl@nomatch}{\textit{No matching files.}}
```

Each recorded file is tested against the filter, and the package's own file is always skipped. Matching uses the kernel's `\in@`, which compares tokens literally and does not expand its arguments, so the name held in `\pfl@file` is expanded into the test with `\expandafter`.

```
\newcommand{\pfl@test}[2]{%
    \in@{printfilelist.sty}{#1}%
    \ifin@\else
        \if\relax\detokenize{#2}\relax
            \pfl@printline
        \else
            \in@{#2}{#1}%
            \ifin@\pfl@printline\fi
        \fi
    \fi
}
```

This prints one entry: in the compact form a comma precedes every entry except the first (tracked by `\ifpfl@found`), while in the detailed form the name is followed by its version, if any, on its own line.

```
\newcommand{\pfl@printline}{%
    \ifpfl@inline
        \ifpfl@found,\space\fi
        \pfl@foundtrue
        \texttt{\pfl@file}%
    \else
        \pfl@foundtrue
        \noindent\texttt{\pfl@file}%
        \ifcsname ver@\pfl@file\endcsname
            : \csname ver@\pfl@file\endcsname%
        \fi
        \par
    \fi
}
```

Change History

v1.0.0

General: Initial version 1

v1.1.0

General: Rewritten to use only the LaTeX kernel, removing the xstring and suffix dependencies; the file list is now read live so the package may be loaded anywhere in the preamble; the starred form prints a compact, inline, comma-separated list of names; a note is printed when no file matches the filter; removed the internal file-list alias 1