The filecontents package*

Scott Pakin
scott+fc@pakin.org

April 2, 2023

Notice

The version of LaTeX released in Fall 2019 incorporates all of this package's functionality (and more) into the LaTeX kernel itself. As a result, there is no longer a need for the `filecontents` package. Please use instead the new, built-in `filecontents` environment. Supply the `overwrite` option to mimic this package's behavior:

```
\begin{filecontents}[overwrite]{⟨filename⟩}
⋮
\end{filecontents}
```

See LaTeX 2ε News, Issue 30 (1-Oct-2019) for the announcement of the new kernel functionality.

1 Introduction

`filecontents` (*env.*) There is a little-known environment called `filecontents` that is built into LaTeX 2ε. Here is `filecontents`' description, which was taken from `ltclass.dtx`:

> The environment `filecontents` is intended for passing the contents of packages, options, or other files along with a document in a single file. It has one argument, which is the name of the file to create. If that file already exists (maybe only in the current directory if the OS supports a notion of a 'current directory' or 'default directory') then nothing happens (except for an information message) and the body of the environment is bypassed. Otherwise, the body of the environment is written verbatim to the file name given as the first argument, together with some comments about how it was produced.
>
> The environment is allowed only before `\documentclass` to ensure that all packages or options necessary for this particular run are present when needed. The begin and end tags should each be on a line by itself. There is also a star-form; this does not write extra comments into the file.

*This file has version number v1.5a, last revised 2023/04/02.

(The comment about `filecontents` being valid only before `\documentclass` is, in fact, untrue. `filecontents` is allowed anywhere in the document's preamble.)

The filecontents package provides a hacked-up version of the `filecontents` and `filecontents*` environments that lifts the two restrictions stated above, namely that existing files are never overwritten and that `filecontents` must be used before the `\documentclass` declaration (really, the `\begin{document}`). filecontents is therefore a more convenient way to write external files from within a LaTeX document than is provided by default by the LaTeX 2_ε kernel.

Sample usage `filecontents` works much like `verbatim`, except that it takes a mandatory filename argument:

```
\begin{filecontents}{myfile.tex}
This text gets written to \texttt{myfile.tex}.
\end{filecontents}
```

The preceding code will write a `myfile.tex` file with contents resembling the following:

```
%% LaTeX2e file `myfile.tex'
%% generated by the `filecontents' environment
%% from source `mydocument' on 2001/07/31.
%%
This text gets written to \texttt{myfile.tex}.
```

`myfile.tex` can then be incorporated back into the document with `\include` or `\input`. Had `filecontents*` been used instead of `filecontents`, the file would have contained only the "`This text gets written to \texttt{myfile.tex}.`" line. `filecontents*` is therefore useful for writing non-LaTeX files such as Encapsulated PostScript files.

If you use the ltxtable package you may find filecontents particularly useful. ltxtable is a crude conglomeration of longtable, which allows tables to cross page boundaries, and tabularx, which enables tables to stretch to a specified width. ltxtable's interface is a bit cumbersome, however; it requires that the `longtable` environment be contained in a separate file. With the filecontents package you can create this file right before the `\LTXtable` invocation—a far more convenient alternative than having to manually place the table within a separate file.

2 Implementation

Most users can stop reading at this point. The Implementation section contains the annotated source code for the filecontents package itself, which is useful only to people who want a detailed and precise explanation of how filecontents works.

To give credit where credit is due, I wrote virtually none of the filecontents code myself. It comes almost exclusively from the LaTeX 2_ε source code, specifically from the file `ltclass.dtx`, which is attributed to Frank Mittelbach, Chris Rowley, Alan Jeffrey, and David Carlisle. I merely made a few small changes (indicated below by

bracketed blocks of code and comments) to make the `filecontents` environment more convenient to use.

```
1 ⟨*package⟩
```

`\filec@ntents@old@kernel` Except where indicated, the source—including comments—to the `\filec@ntents` macro—was taken verbatim from `ltclass.dx`.

```
2 \begingroup%
3 \catcode‘\*=11 %
4 \catcode‘\^^M\active%
5 \catcode‘\^^L\active\let^^L\relax%
6 \catcode‘\^^I\active%
7 \gdef\filec@ntents@old@kernel#1{%
8   \openin\@inputcheck#1 %
```

In the original code a pre-existing file would not be overwritten. In the new version the file existence check is used solely to decide whether to output "`Writing file` ‘⟨*filename*⟩’" or "`Overwriting file` ‘⟨*filename*⟩’". Control flow then always falls through to what used to be the `\ifeof` case (file does not exist; open it), never the `\else` case (file already exists; do nothing).

```
9   \ifeof\@inputcheck%
10    \@latex@warning@no@line%
11        {Writing file ‘\@currdir#1’}%
12  \else %
13    \@latex@warning@no@line%
14        {Overwriting file ‘\@currdir#1’}%
15  \fi %
16  \closein\@inputcheck %
17  \chardef\reserved@c15 %
18  \ch@ck7\reserved@c\write%
19  \immediate\openout\reserved@c#1\relax%
```

```
20  \if@tempswa%
21    \immediate\write\reserved@c{%
22      \@percentchar\@percentchar\space%
23          \expandafter\@gobble\string\LaTeX2e file ‘#1’^^J%
24      \@percentchar\@percentchar\space  generated by the %
25        ‘\@currenvir’ \expandafter\@gobblefour\string\newenvironment^^J%
26      \@percentchar\@percentchar\space from source ‘\jobname’ on %
27         \number\year/\two@digits\month/\two@digits\day.^^J%
28      \@percentchar\@percentchar}%
29  \fi%
30  \let\do\@makeother\dospecials%
```

The inputenc packages might have marked some of the upper 128 character codes "active" (category code 13). That confuses filecontents. Hence, we locally mark each of the upper 128 character codes as "letter" (category code 11) so that they can be written correctly to a file.

```
31  \count0=128\relax %
32  \loop %
```

```
    \catcode\count0=11\relax %
    \advance\count0 by 1\relax %
    \ifnum\count0<256 %
  \repeat %
```

```
  \edef\E{\@backslashchar end\string{\@currenvir\string}}%
  \edef\reserved@b{%
    \def\noexpand\reserved@b%
         ####1\E####2\E####3\relax}%
  \reserved@b{%
    \ifx\relax##3\relax%
```

There was no `\end{filecontents}`

```
      \immediate\write\reserved@c{##1}%
    \else%
```

There was a `\end{filecontents}`, so stop this time.

```
      \edef^^M{\noexpand\end{\@currenvir}}%
      \ifx\relax##1\relax%
      \else%
```

Text before the `\end`, write it with a warning.

```
         \@latex@warning{Writing text `##1' before %
            \string\end{\@currenvir}\MessageBreak as last line of #1}%
        \immediate\write\reserved@c{##1}%
      \fi%
      \ifx\relax##2\relax%
      \else%
```

Text after the `\end`, ignore it with a warning.

```
        \@latex@warning{%
          Ignoring text `##2' after \string\end{\@currenvir}}%
      \fi%
    \fi%
    ^^M}%
```

```
  \catcode`\^^L\active%
  \let\L\@undefined%
  \def^^L{\expandafter\ifx\csname L\endcsname\relax\fi ^^J^^J}%
  \catcode`\^^I\active%
  \let\I\@undefined%
  \def^^I{\expandafter\ifx\csname I\endcsname\relax\fi\space}%
  \catcode`\^^M\active%
  \edef^^M##1^^M{%
    \noexpand\reserved@b##1\E\E\relax}}%
\endgroup
```

`\fc@no@preamblecmds` LaTeX 2_ε declares `\filecontents`, `\filecontents*`, and all of the related helper macros as `\@onlypreamble`, meaning they become invalid after the `\begin{document}`. The following code re-enables their usage anywhere in the document. It was taken from the pkgindoc package (which is generated from `ltclass.dtx`), but modified to re-enable only the commands needed by filecontents, not all of the class and package option-processing commands, as well.

```
\def\fc@no@preamblecmds#1\do\filecontents#2\do\filec@ntents#3\relax{%
  \gdef\@preamblecmds{#1#3}}
```

Newer LaTeX kernels define a most sophisticated `filecontents` environment than what this package provides. We currently use the existence of `\filec@ntents@opt` as indication that the new `filecontents` environment is available.

```
\@ifundefined{filec@ntents@opt}{%
```

Older kernel: Install this package's version of the `filecontents` environment.

```
  \let\filec@ntents=\filec@ntents@old@kernel
  \expandafter\fc@no@preamblecmds\@preamblecmds\relax
}{%
```

Newer kernel: Issue a warning and do not install this package's version of `filecontents`.

```
  \PackageWarningNoLine{filecontents}{%
    This package is obsolete.  Disabling it and\MessageBreak
    passing control to the filecontents environment\MessageBreak
    defined by the LaTeX kernel%
  }%
}
```

⟨/package⟩

Change History

v1.0
General: Initial version 1
v1.1
`\filec@ntents@old@kernel`: Made it possible for filecontents to write Latin-1 characters as per Harry Schmidt's feature request and Frank Mittelbach's suggestion of how to implement it. 3
v1.1a
General: Clarified/corrected the `\documentclass` restriction as per Robin Fairbairns's suggestion 1
v1.2
`\filec@ntents@old@kernel`: Added percent signs after each line in the loop as per Heiko Oberdiek's suggestion for getting filecontents to work with the guitar package 3
Added percent signs after the `\else` and `\fi` lines as per Heiko Oberdiek's suggestion for getting filecontents to work with the guitar package 3
v1.3
`\filec@ntents@old@kernel`: Added a `\closein` to fix bug latex/1487 (reported by Ulrike Fischer and Heiko Oberdiek) . . 3
v1.4
`\filec@ntents@old@kernel`: Update definitions of `^^L` and `^^I` for compatibility with the 2018-04-01 LaTeX release 4
v1.5
General: Declare the package obsolete when running under newer LaTeX kernels 5
v1.5a
General: Documentation updates; no code changes 1

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

Symbols

\@backslashchar . . . 37
\@currdir 11, 14
\@inputcheck . . . 8, 9, 16
\@latex@warning . 48, 54
\@latex@warning@no@line 10, 13
\@percentchar 22, 24, 26, 28
\@preamblecmds . . 70, 73

C

\ch@ck 18

E

environments:
filecontents *1*

F

\fc@no@preamblecmds 69, 73
\filec@ntents . . . 69, 72
\filec@ntents@old@kernel 2, 72
\filecontents 69
filecontents (env.) . . *1*

P

\PackageWarningNoLine 75