

## NAME

**ctanify** – Prepare a package for upload to CTAN

## SYNOPSIS

```
ctanify [--pkgnome=string] [--[no]auto] [--tdsonly=filespec ...] [--[no]unixify]
[--[no]skip] [--tdsdir=dirname ...] [--tex=macro_pkg] [--[no]miscify] [--[no]tds]
filespec[=dirname] ...

ctanify [--help]

ctanify [--version]
```

## DESCRIPTION

**ctanify** is intended for developers who have a LaTeX package that they want to distribute via the Comprehensive TeX Archive Network (CTAN). Given a list of filenames, **ctanify** creates a tarball (a *.tar.gz* file) with the files laid out in CTAN’s preferred structure. The tarball additionally contains a ZIP (*.zip*) file with copies of all files laid out in the standard TeX Directory Structure (TDS), which facilitates inclusion of the package in the TeX Live distribution.

## OPTIONS

**ctanify** accepts the following command-line options:

### **-h, --help**

Output basic usage information and exit.

### **-V, --version**

Output **ctanify**’s version number and exit.

### **-p string, --pkgnome=string**

Specify explicitly a package name. Normally, **ctanify** uses the base name of the first *.ins* or *.sty* file listed as the package name. The package name forms the base name of the tarball that **ctanify** produces.

### **--noauto**

Do not automatically add files to the tarball. Normally, **ctanify** automatically includes all files mentioned in a *.ins* file.

### **-t filespec, --tdsonly=filespec**

Specify a subset of the files named on the command line to include only in the TDS ZIP file, not in the CTAN package directory. Wildcards are allowed (quoted if necessary), and **--tdsonly** can be used multiple times on the same command line.

At least one filename must be specified on the command line. **ctanify** automatically places files in the TDS tree based on their extension, but this can be overridden by specifying explicitly a target TDS directory using the form *filespec=dirname*. Wildcards are allowed for the filespec (quoted if necessary).

## ADDITIONAL OPTIONS

The following options are unlikely to be necessary in ordinary usage. They are provided for special circumstances that may arise.

### **-d dirname, --tdsdir=dirname**

Instead of creating a tarball for CTAN, merely create the package TDS tree rooted in directory *dirname*.

**-T** *macro\_pkg*, **--tex=macro\_pkg**

Assert that the files being packaged for CTAN target a TeX macro package other than LaTeX. Some common examples of *macro\_pkg* are `generic`, `plain`, and `context`.

**-nou, --no-unixify**

Store text files unmodified instead of converting their end-of-line character to Unix format (a single linefeed character with no carriage-return character), even though CTAN prefers receiving all files with Unix-format end-of-line characters.

**-nok, --no-skip**

Force **ctanify** to include files such as Unix hidden files, Emacs backup files, and version-control metadata files, all of which CTAN dislikes receiving.

**-m, --miscify**

Rename directories containing a single file to `misc`. (For example, rename `tex/latex/mypackage/mypackage.sty` to `tex/latex/misc/mypackage.sty`.) This was common practice in the past but is now strongly discouraged.

**-nos, --no-tds**

Do not embed a `.tds.zip` file in the generated tarball.

**DIAGNOSTICS**

Failed to copy *filename* (No such file or directory)

This message is typically caused by a `.ins` file that generates *filename* but that has not already been run through `tex` or `latex` to actually produce *filename*. **ctanify** does not automatically run `tex` or `latex`; this needs to be done manually by the user. See “CAVEATS” for more information.

Modified *filename* to use Unix line endings (use `--no-unixify` to prevent this)

For consistency, CTAN stores all text files with Unix-style line endings (a single linefeed character with no carriage-return character). To help in this effort, **ctanify** automatically replaces non-Unix-style line endings. The preceding merely message notifies the user that he should not be alarmed to see a different size for *filename* in the tarball versus the original *filename* on disk (which **ctanify** never modifies). If there’s a good reason to preserve the original line endings (and there rarely is), the `--no-unixify` option can be used to prevent **ctanify** from altering any files when storing them in the tarball.

Excluding *filename* (use `--no-skip` to force inclusion)

**ctanify** normally ignores files—even when specified explicitly on the command line—that CTAN prefers not receiving. These include files whose names start with “.” (Unix hidden files), end in “~” (Emacs automatic backups), or that come from a `CVS` or `.svn` directory (version-control metadata files). If there’s a good reason to submit such files to CTAN (and there rarely is), the `--no-skip` option can be used to prevent **ctanify** from ignoring them.

CTAN prefers having only PDF documentation (re: *filename*)

Because of the popularity of the PDF format, CTAN wants to have as much documentation as possible distributed in PDF. The preceding message asks the user to replace any PostScript or DVI documentation with PDF if possible. (**ctanify** will still include PostScript and DVI documentation in the tarball; the preceding message is merely a polite request.)

Not including *filename* in the TDS tree (unknown extension)  
**ctanify** places files in the TDS tree based on a table of file extensions. For example, all *.sty* files are placed in *tex/latex/package-name*. If **ctanify** does not know where to put a file it does not put it anywhere. See the last paragraph of “OPTIONS” for an explanation of how to specify explicitly a file’s target location in the TDS tree. For common file extensions that happen to be absent from **ctanify**’s table, consider also notifying **ctanify**’s author at the address shown below under “AUTHOR”.

## EXAMPLES

### The Common Case

Normally, all that’s needed is to tell **ctanify** the name of the *.ins* file (or *.sty* if the package does not use DocStrip) and the prebuilt documentation, if any:

```
$ ctanify mypackage.ins mypackage.pdf README
490347 mypackage.tar.gz

1771 mypackage/README
15453 mypackage/mypackage.dtx
1957 mypackage/mypackage.ins
277683 mypackage/mypackage.pdf
246935 mypackage.tds.zip

1771 doc/latex/mypackage/README
277683 doc/latex/mypackage/mypackage.pdf
15453 source/latex/mypackage/mypackage.dtx
1957 source/latex/mypackage/mypackage.ins
1725 tex/latex/mypackage/mypackage.sty
```

**ctanify** outputs the size in bytes of the resulting tarball, each file within it, and each file within the contained ZIP file. In the preceding example, notice how **ctanify** automatically performed all of the following operations:

- including *mypackage.dtx* (found by parsing *mypackage.ins*) in both the *mypackage* directory and the ZIP file,
- including *mypackage.sty* (found by parsing *mypackage.ins*) in the ZIP file but, because it’s a generated file, not in the *mypackage* directory, and
- placing all files into appropriate TDS directories (documentation, source, main package) within the ZIP file.

Consider what it would take to manually produce an equivalent *mypackage.tar.gz* file. **ctanify** is definitely a simpler, quicker alternative.

### Advanced Usage

**ctanify** assumes that PostScript files are documentation and therefore stores them under *doc/latex/package-name/* in the TDS tree within the ZIP File. Suppose, however, that a LaTeX package uses a set of PostScript files to control **dvips**’s output. In this case, **ctanify** must be told to include those PostScript files in the package directory, not the documentation directory.

```
$ ctanify mypackage.ins "mypackage*.ps=tex/latex/mypackage"
```

## FILES

*perl* **ctanify** is written in Perl and needs a Perl installation to run.

*tar, gzip*

**ctanify** requires the GNU *tar* and *gzip* programs to create a compressed tarball (*.tar.gz*).

*zip* **ctanify** uses a *zip* program to archive the TDS tree within the main tarball.

## CAVEATS

**ctanify** does not invoke *tex* or *latex* on its own, e.g., to process a *.ins* file. The reason is that **ctanify** does not know in the general case how to produce all of a package's generated files. It was deemed better to do nothing than to risk overwriting existing *.sty* (or other) files or to include outdated generated files in the tarball. In short, before running **ctanify** you should manually process any *.ins* files and otherwise generate any files that should be sent to CTAN.

**ctanify** has been tested only on Linux. It may work on OS X. I've been told that it works on Windows when run using Cygwin. Volunteers willing to help port **ctanify** to other platforms are extremely welcome.

## SEE ALSO

*tar*(1), *zip*(1), *latex*(1), Guidelines for uploading TDS-Packaged materials to CTAN (<<http://www.ctan.org/TDS-guidelines>>), A Directory Structure for TeX Files (<<http://tug.org/tds/>>),

## AUTHOR

Scott Pakin, [scott+ctnfy@pakin.org](mailto:scott+ctnfy@pakin.org)

## COPYRIGHT AND LICENSE

Copyright 2017 Scott Pakin

This work may be distributed and/or modified under the conditions of the LaTeX Project Public License, either version 1.3c of this license or (at your option) any later version. The latest version of this license is in

<<http://www.latex-project.org/lppl.txt>>

and version 1.3c or later is part of all distributions of LaTeX version 2008/05/04 or later.