

**NAME**

dos2unix – DOS/MAC to UNIX and vice versa text file format converter

**SYNOPSIS**

```
dos2unix [options] [-c CONVMODE] [-o FILE ...] [-n INFILE OUTFILE ...]
unix2dos [options] [-c CONVMODE] [-o FILE ...] [-n INFILE OUTFILE ...]
```

**DESCRIPTION**

The Dos2unix package includes utilities `dos2unix` and `unix2dos` to convert plain text files in DOS or MAC format to UNIX format and vice versa. Binary files and non-regular files, such as soft links, are automatically skipped, unless conversion is forced.

Dos2unix has a few conversion modes similar to `dos2unix` under SunOS/Solaris.

In DOS/Windows text files line endings exist out of a combination of two characters: a Carriage Return (CR) followed by a Line Feed (LF). In Unix text files line endings exists out of a single Newline character which is equal to a DOS Line Feed (LF) character. In Mac text files, prior to Mac OS X, line endings exist out of a single Carriage Return character. Mac OS X is Unix based and has the same line endings as Unix.

**OPTIONS**

**-c, --convmode CONVMODE**

Set conversion mode. Where CONVMODE is one of: *ascii*, *7bit*, *iso*, *mac* with *ascii* being the default.

**-f, --force**

Force conversion of all files. Also binary files.

**-h, --help**

Display online help.

**-k, --keepdate**

Keep the date stamp of output file same as input file.

**-L, --license**

Display software license.

**-l, --newline**

Add additional newline.

**dos2unix:** Only DOS line endings are changed to two Unix line endings. In Mac mode only Mac line endings are changed to two Unix line endings.

**unix2dos:** Only Unix line endings are changed to two DOS line endings. In Mac mode Unix line endings are changed to two Mac line endings.

**-n, --newfile INFILE OUTFILE ...**

New file mode. Convert the infile and write output to outfile. File names must be given in pairs and wildcard names should NOT be used or you WILL lose your files.

**-o, --oldfile FILE ...**

Old file mode. Convert the file and write output to it. The program default to run in this mode. Wildcard names may be used.

**-q, --quiet**

Quiet mode. Suppress all warning and messages.

**-V, --version**

Display version information.

**CONVERSION MODES**

Conversion modes *ascii*, *7bit*, and *iso* are similar to those of `dos2unix`/`unix2dos` under SunOS/Solaris.

**ascii**

**dos2unix:** In this mode DOS line endings are converted to Unix line endings. Unix and Mac line endings are not changed.

**unix2dos:** In this mode Unix line endings are converted to DOS line endings. DOS and Mac line

endings are not changed.

Although the name of this mode is ASCII, which is a 7 bit standard, the actual mode is 8 bit.

#### mac

**dos2unix:** In this mode Mac line endings are converted to Unix line endings. DOS and Unix line endings are not changed. You can also use the command `mac2unix` to run `dos2unix` in Mac mode.

**unix2dos:** In this mode Unix line endings are converted to Mac line endings. DOS and Mac line endings are not changed. You can also use the command `unix2mac` to run `unix2dos` in Mac mode.

#### 7bit

In this mode DOS line endings are converted to Unix line endings or vice versa. All 8 bit non-ASCII characters (with values from 128 to 255) are converted to a space.

**iso** In this mode DOS line endings are converted to Unix line endings or vice versa. Characters are converted between the DOS character set (code page) CP437 and ISO character set ISO-8859-1 on Unix. CP437 characters without ISO-8859-1 equivalent, for which conversion is not possible, are converted to a dot. The same counts for ISO-8859-1 characters without CP437 counterpart. CP437 is mainly used in the USA. In Western Europe CP850 is more standard.

Another option to convert text files between different encodings is to use `dos2unix` in combination with `iconv(1)`. `Iconv` can convert between a long list of character encodings. Some examples:

Convert from DOS DOSLatinUS to Unix Latin-1

```
iconv -f CP437 -t ISO-8859-1 in.txt | dos2unix > out.txt
```

Convert from DOS DOSLatin1 to Unix Latin-1

```
iconv -f CP850 -t ISO-8859-1 in.txt | dos2unix > out.txt
```

Convert from Windows WinLatin1 to Unix Latin-1

```
iconv -f CP1252 -t ISO-8859-1 in.txt | dos2unix > out.txt
```

Convert from Windows WinLatin1 to Unix UTF-8 (Unicode)

```
iconv -f CP1252 -t UTF-8 in.txt | dos2unix > out.txt
```

Convert from Windows UTF-16 (Unicode) to Unix UTF-8 (Unicode)

```
iconv -f UTF-16 -t UTF-8 in.txt | dos2unix > out.txt
```

Convert from Unix Latin-1 to DOS DOSLatinUS

```
unix2dos < in.txt | iconv -f ISO-8859-1 -t CP437 > out.txt
```

Convert from Unix Latin-1 to DOS DOSLatin1

```
unix2dos < in.txt | iconv -f ISO-8859-1 -t CP850 > out.txt
```

Convert from Unix Latin-1 to Windows WinLatin1

```
unix2dos < in.txt | iconv -f ISO-8859-1 -t CP1252 > out.txt
```

Convert from Unix UTF-8 (Unicode) to Windows WinLatin1

```
unix2dos < in.txt | iconv -f UTF-8 -t CP1252 in.txt > out.txt
```

Convert from Unix UTF-8 (Unicode) to Windows UTF-16 (Unicode)

```
unix2dos < in.txt | iconv -f UTF-8 -t UTF-16 > out.txt
```

See also <http://czyborra.com/charsets/codepages.html> and <http://czyborra.com/charsets/iso8859.html>.

## UNICODE

Unicode files can be encoded in different encodings. On Unix/Linux Unicode files are mostly encoded in UTF-8 encoding. UTF-8 is ASCII compatible. UTF-8 files can be in DOS, Unix or Mac format. It is safe to run dos2unix/unix2dos on UTF-8 encoded files. On Windows mostly UTF-16 encoding is used for Unicode files. Dos2unix/unix2dos should not be run on UTF-16 files. UTF-16 files are automatically skipped, because it are binary files.

## EXAMPLES

Get input from stdin and write output to stdout.

```
dos2unix
dos2unix -l -c mac
```

Convert and replace a.txt. Convert and replace b.txt.

```
dos2unix a.txt b.txt
dos2unix -o a.txt b.txt
```

Convert and replace a.txt in ascii conversion mode.

```
dos2unix a.txt
```

Convert and replace a.txt in ascii conversion mode. Convert and replace b.txt in 7bit conversion mode.

```
dos2unix a.txt -c 7bit b.txt
dos2unix -c ascii a.txt -c 7bit b.txt
```

Convert a.txt from Mac to Unix format.

```
dos2unix -c mac a.txt
mac2unix a.txt
```

Convert a.txt from Unix to Mac format.

```
unix2dos -c mac a.txt
unix2mac a.txt
```

Convert and replace a.txt while keeping original date stamp.

```
dos2unix -k a.txt
dos2unix -k -o a.txt
```

Convert a.txt and write to e.txt.

```
dos2unix -n a.txt e.txt
```

Convert a.txt and write to e.txt, keep date stamp of e.txt same as a.txt.

```
dos2unix -k -n a.txt e.txt
```

Convert and replace a.txt. Convert b.txt and write to e.txt.

```
dos2unix a.txt -n b.txt e.txt
dos2unix -o a.txt -n b.txt e.txt
```

Convert c.txt and write to e.txt. Convert and replace a.txt. Convert and replace b.txt. Convert d.txt and write to f.txt.

```
dos2unix -n c.txt e.txt -o a.txt b.txt -n d.txt f.txt
```

## LOCALIZATION

### LANG

The primary language is selected with the environment variable LANG. The LANG variable consists out of several parts. The first part is in small letters the language code. The second is optional and is the country code in capital letters, preceded with an underscore. There is also an optional third part: character encoding, preceded with a dot. A few examples for POSIX standard type shells:

```

export LANG=nl                Dutch
export LANG=nl_NL             Dutch, The Netherlands
export LANG=nl_BE             Dutch, Belgium
export LANG=es_ES             Spanish, Spain
export LANG=es_MX             Spanish, Mexico
export LANG=en_US.iso88591    English, USA, Latin-1 encoding
export LANG=en_GB.UTF-8       English, UK, UTF-8 encoding

```

For a complete list of language and country codes see the gettext manual:  
[<http://www.gnu.org/software/gettext/manual/gettext.html#Language-Codes>](http://www.gnu.org/software/gettext/manual/gettext.html#Language-Codes)

On Unix systems you can use to command *locale* (1) to get locale specific information.

## LANGUAGE

With the LANGUAGE environment variable you can specify a priority list of languages, separated by colons. Dos2unix gives preference to LANGUAGE over LANG. For instance, first Dutch and then German: LANGUAGE=nl:de. You have to first enable localization, by setting LANG (or LC\_ALL) to a value other than “C”, before you can use a language priority list through the LANGUAGE variable. See also the gettext manual:  
[<http://www.gnu.org/software/gettext/manual/gettext.html#The-LANGUAGE-variable>](http://www.gnu.org/software/gettext/manual/gettext.html#The-LANGUAGE-variable)

For Esperanto there is a special language file in x-method format. X-method can be used on systems that don’t support Latin-3 or Unicode character encoding. Make LANGUAGE equal to “eo-x:eo”.

If you select a language which is not available you will get the standard English messages.

## DOS2UNIX\_LOCALEDIR

With the environment variable DOS2UNIX\_LOCALEDIR the LOCALEDIR set during compilation can be overruled. LOCALEDIR is used to find the language files. The GNU default value is /usr/local/share/locale. Option “-V” will display the LOCALEDIR that is used.

Example (windows cmd):

```
set DOS2UNIX_LOCALEDIR=c:/my_prefix/share/locale
```

## AUTHORS

Benjamin Lin – <blin@socs.uts.edu.au>

Bernd Johannes Wuebben (mac2unix mode) – <wuebben@kde.org>

Erwin Waterlander – <waterlan@xs4all.nl>

Project page: <<http://www.xs4all.nl/~waterlan/dos2unix.html>>

SourceForge page: <<http://sourceforge.net/projects/dos2unix/>>

Freshmeat: <<http://freshmeat.net/projects/dos2unix>>

## SEE ALSO

*iconv* (1)