# CSV Processing With gawk

A User's Guide for the CSV extension of GNU Awk

Edition 1.0

March, 2018

## Manuel Collado

Published by:

Free Software Foundation
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335 USA
Phone: +1-617-542-5942
Fax: +1-617-542-2652
Email: `gnu@gnu.org`
URL: `http://www.gnu.org/`

This is Edition 1.0 of *CSV Processing With* `gawk`, for the 1.0.0 (or later) version of the CSV extension of the GNU implementation of AWK.

# Table of Contents

1 Introduction .................................... 1

2 CSV Extension Tutorial ........................ 3

2.1 The CSV data format ........................................ 3

2.2 Installing the CSV extension ............................... 3

Prerequisites ................................................ 3

Download the sources ......................................... 3

Compile the sources .......................................... 3

2.3 Using the CSV extension .................................... 3

Parsing individual CSV records ............................... 4

Automatic parsing of CSV files ............................... 5

Generating CSV data .......................................... 7

3 CSV Extension Reference ..................... 9

3.1 CSV parse functions ....................................... 10

NAME ........................................................ 10

USAGE ....................................................... 10

DESCRIPTION ................................................. 10

EXAMPLES .................................................... 11

NOTES ....................................................... 11

LIMITATIONS ................................................. 11

3.2 CSV input mode ............................................ 12

NAME ........................................................ 12

USAGE ....................................................... 12

DESCRIPTION ................................................. 12

EXAMPLES .................................................... 13

NOTES ....................................................... 13

LIMITATIONS ................................................. 13

3.3 CSV data generation ....................................... 14

NAME ........................................................ 14

USAGE ....................................................... 14

DESCRIPTION ................................................. 14

EXAMPLES .................................................... 15

NOTES ....................................................... 15

LIMITATIONS ................................................. 15

Appendix A CSV Specification ................. 17

Appendix B GNU Free Documentation License .. 19

Index ............................................. 27

# 1 Introduction

The CSV extension of gawk provides facilities for handling input and output CSV formatted data.

On input, CSV records can be processed individually. There are CSV parsing functions that can extract field values from a CSV record or convert the CSV record into a plain text record with fixed field delimiters.

It is also possible to process whole CSV data files by automatically reading and converting each CSV record and delivering it as $0, $1, .. $NF, as if it were a simple text tabular data file.

On output, CSV formatted records can be generated from either an array of field values or from a simple text record with fixed field delimiters.

The CSV format is not well standardized. The gawk CSV extension can handle cvs-like data with custom field delimiter and quoting characters.

# 2 CSV Extension Tutorial

## 2.1 The CSV data format

The Comma-Separated-Values (CSV) data format is commonly used by spreadsheets and database engines to import and export data as plain text files.

A CSV file is a sequence of records separated by newline marks. A CSV record is a sequence of fields separated by commas. A field can contain almost any text. If a field contains commas, newlines or double quotes it must be enclosed in double quotes. Double quotes inside a field must be escaped by doubling them. Example:

```
author,title,remarks
Shakespeare,A Midsummer Night’s Dream,comedy
"Stevenson, Robert Louis",Treasure Island,novel
anonymous,"A ""quoted"" word","remark 1
remark 2"
```

There are four records, each one with three fields. The field "Stevenson, Robert Louis" is quoted because it contains a comma. The field "A ""quoted"" word" is quoted because it contains escaped quotes (coded as duplicates). The third field of last record has two lines of text. The data is equivalent to the following table:

| **author** | **title** | **remarks** |
|---|---|---|
| Shakespeare | A Midsummer Night’s Dream | comedy |
| Stevenson, Robert Louis | Treasure Island | novel |
| anonymous | A "quoted" word | remark 1<br>remark 2 |

## 2.2 Installing the CSV extension

The gawk-csv extension is distributed mainly as source code.

### Prerequisites

The gawk-csv extension requires:

- GNU awk (gawk) version 4.1.1 or later.
- The gawkextlib common base library.

### Download the sources

From the gawkextlib project at SourceForge.

### Compile the sources

./configure && make && make check && make install.

## 2.3 Using the CSV extension

The gawk-csv extension provides facilities for:

- Process strings containing individual CSV records.

- Process whole CSV data files.
- Generate CSV formatted records.

The gawk-csv extension must be explicitly loaded with either a **-i csv** option in the command line or a **@include "csv"** directive in the awk script code.

## Parsing individual CSV records

The `csvsplit()` function can extract the field values from a CSV formatted record string. The field values are stored as elements of an array. Example:

```
data --> a,"b,c",d
n = csvsplit(data, af)
```

gives

```
n = 3
af[1] = "a"
af[2] = "b,c"
af[3] = "d"
```

It is possible to handle data that use alternate delimiter or quote characters. For instance, if the record uses semicolons instead of commas to delimit fields, and single quotes instead of double quotes:

```
data --> a;’b;c’;d
n = csvsplit(data, af, ";", "’")
```

gives

```
n = 3
af[1] = "a"
af[2] = "b;c"
af[3] = "d"
```

Another possibility is to use the `csvconvert()` function. It converts a CSV record into a simple record with fields delimited by a fixed text given as argument. Example:

```
data --> a,"b,c",d
str = csvconvert(data, "|")
```

gives

```
str = "a|b,c|d"
```

The `csvconvert()` function also accepts alternate delimiter or quoting characters:

```
data --> a;’b;c’;d
str = csvconvert(data, "|", ";", "’")
```

gives

```
str = "a|b;c|d"
```

Of course, the fixed field delimiter of the converted record should not appear as data inside the CSV record. Otherwise the data structure will be fouled up. By default, `csvconvert()` uses null characters as field delimiters in the converted record. This seem a convenient option, because CSV data are not expected to contain null characters:

```
data --> a,"b,c",d
str = csvconvert(data)
```

gives

```
str = "a\0b,c\0d"
```

## Automatic parsing of CSV files

Automatic parsing of CSV data files is controlled by a predefined CSVMODE control variable. If set to `1` the input data file reader automatically recognizes CSV records and splits them into fields as expected. The fields are delivered as $1, $2, ... $NF as usual.

Sample data file:

```
a,b,c
p,"q,r",s
x,"""y""",z
```

Awk script:

```
@include "csv"
BEGIN { CSVMODE = 1 }
{ print $2 }
```

Result:

```
b
q,r
"y"
```

The parsing process can be customized in order to accept non-standard CSV data files. A couple of predefined variables can be used to specify special field delimiter and quoting characters:

CSVCOMMA: The special character that delimit the fields. By default a comma (’,’).

CSVQUOTE: The specific character used to quote values. By default a double quote (").

Sample data file:

```
a;b;c
p;q,r;s
x;’"y"’;z
```

Awk script:

```
@include "csv"
BEGIN { CSVMODE = 1; CSVCOMMA = ";"; CSVQUOTE = "’" }
{ print $2 }
```

Result:

```
b
q,r
"y"
```

The whole CSV record is stored as $0. Not in its original form, but as the concatenation of the fields, now delimited by a fixed separator. By default this separator is the null character (’\0’). The user can change it by means of the CSVFS predefined variable. It is the user responsibility to use a value that cannot appear inside the CSV data.

Sample data file:

```
a,b,c
```

```
p,"q,r",s
x,"""y""",z
```

Awk script:

```
@include "csv"
BEGIN { CSVMODE = 1; CSVFS = "|" }
{ print }
```

Result:

```
a|b|c
p|q,r|s
x|"y"|z
```

File processing in the automatic CSVMODE correctly recognizes CSV records with multiline fields. I.e., fields that contain newline characters.

Sample data file:

```
a,b,c
p,"q
r",s
x,"""y""",z
```

Awk script:

```
@include "csv"
BEGIN { CSVMODE = 1 }
{ print "<" $2 ">" }
```

Result:

```
<b>
<q
r>
<"y">
```

Even if the automatic parsing of CSV files rebuilds the record, the original representation is not lost. The predefined CSVRECORD variable holds this original value. It is really easy to extract selected records of a CSV file:

Sample data file:

```
a,b,c
p,"q,r",s
p,"a,r",s
x,"""y""",z
x,"""a""",z
```

Awk script:

```
@include "csv"
BEGIN { CSVMODE = 1 }
# Extract records that contain 'a' in the second field
$2 ~ /a/ { print CSVRECORD }
```

Result:

```
p,"a,r",s
x,"""a""",z
```

## Generating CSV data

In addition to capabilities for reading or converting CSV input data records, the gawk-csv extension also provides facilities for creating CSV records. These facilities are implemented by an awk library called `csv.awk`, that must be explicitly included with either a **-i csv** option in the command line or a **@include** `"csv"` directive in the awk script code.

A CSV record can be created two ways:

- From an array of fields.
- From a regular record string with fields delimited by a FS-like pattern.

**csvcompose**(*afield* [, *comma* [, *quote*]])

Returns a CSV formatted string by composing the values in the **afield** array, indexed from 1 to *N*. The optional *comma* argument is the desired field delimiter, by default a comma (,). And the optional *quote* argument is the desired quoting character, by default a double quote (").

Example:

```
f[1] = "007"
f[2] = "Bond, James"
f[3] = "United Kingdom"
result = csvcompose(f)  # -> '007,"Bond, James",United Kingdom'
result = csvcompose(f, ";")  # -> '007;Bond, James;United Kingdom'
```

**csvformat**(*record*, [*fs* [, *comma* [, *quote*]]])

Returns a CSV formatted string by recomposing the fields in the **record** string. The optional **fs** argument is the field separator pattern used in the *record* argument, by default a null character (\0). The optional *comma* and *quote* arguments are the same as the *csvcompose()* function ones.

Example:

```
record = "007/Bond, James/United Kingdom"
result = csvformat(record, "/")  # -> '007,"Bond, James",United Kingdom'
result = csvformat(record, "/", ";")  # -> '007;Bond, James;United Kingdom'
```

# 3 CSV Extension Reference

This chapter is meant to be a reference. It collects the manual pages that describe each feature or group of features. These manual pages are also available separately. The first two sections describe builtin features, while the third describes facilities implemented as awk code library.

## 3.1 CSV parse functions

### NAME

csvconvert, csvsplit - facilities for parsing Comma-Separated-Values (CSV) data with gawk.

### USAGE

```
@include "csv"
...
CSVFS = ...
CSVCOMMA = ...
CSVQUOTE = ...
...
result = csvconvert(csvrecord, option...)
n = csvsplit(csvrecord, afield, option...)
result = csvunquote(csvfield, option)      (see NOTE 1)
```

### DESCRIPTION

The *csv* gawk extension adds functions for parsing CSV data in a simple way. The predefined `CSVFS`, `CSVCOMMA` and `CSVQUOTE` variables set default values for the optional arguments.

**CSVFS** The field delimiter used in the resulting clean text record, initialized to a null character ’\0’.

**CSVCOMMA**
The default field delimiter of the CSV input text, initialized to comma ’,’.

**CSVQUOTE**
The default quoting character of the CSV input text, initialized to double quote ’"’.

**csvconvert(***csvrecord* [, *fs* [, *comma* [, *quote*]]]**)**
Returns the CSV formatted string argument converted to a regular awk record with fixed field separators. Returns a null string if *csvrecord* is not a valid string. The arguments are as follows:

**csvrecord** The CSV formatted input string

**fs** The resulting field separator. Default `CSVFS`.

**comma** The input CSV field delimiter. Default `CSVCOMMA`.

**quote** The input CSV quoting character. Default `CSVQUOTE`.

**csvsplit(***csvrecord*, *afield* [, *comma* [, *quote*]]]**)**
Splits the CSV formatted string argument into an array of individual clean text fields and returns the number of fields. Returns -1 if *csvrecord* is not a valid string. The arguments are as follows:

**csvrecord** The CSV formatted input string

**afield** The resulting array of fields.

**comma** The input CSV field delimiter. Default `CSVCOMMA`.

**quote** The input CSV quoting character. Default `CSVQUOTE`.

**csvunquote(***csvfield* [, *quote*])

Returns the clean text value of the CSV string argument. Returns a null string if *csvfield* is not a valid string. The arguments are as follows:

**csvfield** The CSV formatted input string

**quote** The input CSV quoting character. Default `CSVQUOTE`.

## EXAMPLES

Process CSV input records as arrays of fields:

```
{
    csvsplit($0, fields)
    if (fields[2]=="some value") print
}
```

Process CSV input records as awk regular records:

```
BEGIN {FS = "\0"}
{
    CSVRECORD = $0
    $0 = csvconvert($0)
    if ($2=="some value") print CSVRECORD
}
```

## NOTES

## LIMITATIONS

Null characters are not allowed in fields. A null character terminates the record processing.

## 3.2 CSV input mode

### NAME

csvmode - direct processing of Comma-Separated-Values (CSV) data files with gawk.

### USAGE

```
@include "csv"
BEGIN { CSVMODE = 1 }
  ... rules with $0, $1, ... $NF, CSVRECORD, ...

csvfield(name, default)
csvprint(record, option...)
csvprint0()
```

### DESCRIPTION

The *gawk-csv* extension can directly process CSV data files. Uses some specific variables:

**CSVMODE**

Setting **CSVMODE=1** lets CSV formatted input data records to be automatically converted to regular awk records with fixed field separators, and delivered as **$0**. And **$1** .. **$NF** are also set accordingly. Setting **CSVMODE=0** disables the conversion, and input files are processed the usual way. See NOTE 1.

The conversion can be customized by some control variables:

**CSVFS** The resulting field separator, that temporarily overrides the `FS` and `OFS` predefined variables. If not set, a null char '\0' is used. See NOTE 1.

**CSVCOMMA**

The input CSV field delimiter. Default comma ','.

**CSVQUOTE**

The input CSV quoting character. Default double quote '`"`'.

**CSVRECORD**

The original CSV input record.

If the CSV file has a header record, the fields can also be accessed by name:

**csvfield(***name* [, *missing*])

Returns the named field of the current record. If there is no column named *name*, then return *missing*, or a null value if not given.

**csvprint(**[*record*, [*fs* [, *comma* [, *quote*]]]])

A convenience function to format and print the given record with a single call. If called without arguments it prints either $0 formatted as CSV or `CSVRECORD`, depending on `CSVMODE`. Arguments are like `csvformat()`.

**csvprint0()**

A convenience function to print the original input record as such. Prints either $0 or `CSVRECORD`, depending on `CSVMODE`.

`CSVMODE`, `CSVFS`, `CSVCOMMA` and `CSVQUOTE` are checked only at `BEGINFILE` time. Changing them in the middle of a file processing takes no effect.

`CSVRECORD` is updated for each CSV input record.

The CSV input mode accepts fields with embedded newlines, tabs and other control characters, except null characters ('\0').

## EXAMPLES

Extract CSV records with some specific value in the second field:

```
BEGIN {CSVMODE = 1}
$2=="some value" {print CSVRECORD}
```

Process CSV files with fields separated by semicolons instead of commas:

```
BEGIN {CSVMODE = 1; CSVFS = ";"}
  ... processing rules ...
```

Print a specific named field of every record:

```
BEGIN {CSVMODE = 1;}
{ print csvfield("City") }
```

Print records that contain commas as data, in both normal and CSV modes:

```
grepcommas.awk:
BEGINFILE {
    CSVMODE = (FILENAME ~ /\.csv$/)
}
/,/ { csvprint0() }

Sample invocation:
gawk -f grepcommas.awk a.txt, b.csv, c.txt
```

## NOTES

**(1)** If the user code has a `BEGINFILE` action that sets `CSV-`*`mode`* variables depending on the current file, this action must appear before the `@include "csv"` clause:

```
BEGINFILE {
    CSVMODE = (FILENAME ~ /\.csv$/)  # switch mode depending on the file type
}
@include "csv"
```

## LIMITATIONS

Null characters are not allowed in fields. A null character terminates the record processing.

## 3.3 CSV data generation

### NAME

csv - facilities for creating Comma-Separated-Values (CSV) data with gawk.

### USAGE

```
@include "csv"
...
result = csvcompose(afield, option...)
result = csvformat(record, option...)
result = csvquote(field, option...)
```

### DESCRIPTION

The `csv.awk` library provides control variables and functions for composing CSV data records and fields:

**CSVFS** The expected field separator in the clean text record to be formatted. Default the null character '\0'.

**CSVCOMMA**
The resulting CSV field delimiter. Default comma ','.

**CSVQUOTE**
The resulting CSV quoting character. Default double quote '"'.

**csvcompose(***afield* [, *comma* [, *quote*]]**)**
Returns a CSV formatted string by composing the values in the *afield* array. The arguments are as follows:

**afield** An array of field values, indexed from 1 to *N*.

**comma** Optional. The resulting CSV field delimiter. Default `CSVCOMMA`.

**quote** Optional. The resulting CSV quoting character. Default `CSVQUOTE`.

**csvformat(***record*, [*fs* [, *comma* [, *quote*]]]**)**
Returns a CSV formatted string by composing the fields in the *record* string. The arguments are as follows:

**record** A string record with fields delimited by *fs*.

**fs** Optional. The actual field separator in *record*. Default `CSVFS`.

**comma** Optional. The desired CSV field delimiter. Default `CSVCOMMA`.

**quote** Optional. The desired CSV quoting character. Default `CSVQUOTE`.

**csvquote(***field* [, *comma* [, *quote*]]**)**
Returns a CSV formatted string by escaping the required characters in the *field* string. The arguments are as follows:

**field** A single field clean text string.

**comma** Optional. The desired CSV field delimiter. Default `CSVCOMMA`.

**quote** Optional. The desired CSV quoting character. Default `CSVQUOTE`.

## EXAMPLES

Explicit CSV composition:

```
f[1] = "007"
f[2] = "Bond, James"
f[3] = "United Kingdom"
result = csvcompose(f)  # -> '007,"Bond, James",United Kingdom'
result = csvcompose(f, ";")  # -> '007;Bond, James;United Kingdom'

record = "007/Bond, James/United Kingdom"
result = csvformat(record, "/")  # -> '007,"Bond, James",United Kingdom'
result = csvformat(record, "/", ";")  # -> '007;Bond, James;United Kingdom'
```

## NOTES

The *csv* library automatically loads the CSV extension.

## LIMITATIONS

# Appendix A CSV Specification

The term CSV means "Comma-Separated Values". It is a plain text format usually used by spreadsheets and database engines for interchange of information. In spite of been widely used, the CSV file format is not formally standardized. A commonly used definition is RFC 4180 (`https://tools.ietf.org/html/rfc4180`).

RFC 4180 is quite strict. In practice CSV aware tools accept or generate files not strictly conformant with this specification. Usual deviations are:

- Line endings: LF alone instead of CR+LF.
- Field delimiter character: semicolon or other specific character instead of comma.
- Quoting character: single quote or other specific character instead of double quote.
- Control characters other that line breaks allowed inside field contents.
- Leading and trailing space in fields are not significant and can be ignored.
- Etc.

# Appendix B GNU Free Documentation License

Version 1.3, 3 November 2008

0. PREAMBLE

   The purpose of this License is to make a manual, textbook, or other functional and useful document *free* in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

   This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

   We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

   This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

   A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

   A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

   The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

The "publisher" means any person or entity that distributes copies of the Document to the public.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.

C. State on the Title page the name of the publisher of the Modified Version, as the publisher.

D. Preserve all the copyright notices of the Document.

E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

H. Include an unaltered copy of this License.

I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.

M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.

N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.

O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

   You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

   The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

   In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements."

6. COLLECTIONS OF DOCUMENTS

   You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

   You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

   A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

   If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

   Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

   If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

   You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

   However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

   Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

   Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

10. FUTURE REVISIONS OF THIS LICENSE

    The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See `http://www.gnu.org/copyleft/`.

    Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

11. RELICENSING

    "Massive Multiauthor Collaboration Site" (or "MMC Site") means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A "Massive Multiauthor Collaboration" (or "MMC") contained in the site means any set of copyrightable works thus published on the MMC site.

    "CC-BY-SA" means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

    "Incorporate" means to publish or republish a Document, in whole or in part, as part of another Document.

    An MMC is "eligible for relicensing" if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

    The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.

## ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

```
Copyright (C)  year  your name.
Permission is granted to copy, distribute and/or modify this document
under the terms of the GNU Free Documentation License, Version 1.3
or any later version published by the Free Software Foundation;
with no Invariant Sections, no Front-Cover Texts, and no Back-Cover
Texts.  A copy of the license is included in the section entitled ``GNU
Free Documentation License''.
```

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the “with. . . Texts.” line with this:

```
with the Invariant Sections being list their titles, with
the Front-Cover Texts being list, and with the Back-Cover Texts
being list.
```

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

# Index

## C

Comma-Separated Values . . . . . . . . . . . . . . . . . . . . . . 17
CSVCOMMA . . . . . . . . . . . . . . . . . . . . . . . . 5, 10, 12, 14
csvcompose . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 7, 14
csvconvert . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 4, 10
csvfield . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 12
csvformat . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 7, 14
CSVFS . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 5, 10, 12, 14
CSVMODE . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 5, 12
csvprint . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 12
csvprint0 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 12
csvquote . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 14
CSVQUOTE . . . . . . . . . . . . . . . . . . . . . . . . . 5, 10, 12, 14
CSVRECORD . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6, 12
csvsplit . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 4, 10
csvunquote . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 11

## F

FDL, GNU Free Documentation License . . . . . . . . 19

## M

multiline fields . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6

## R

RFC 4180 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 17