

# GNU Web Translators Manual

---

Documentation for translators of [www.gnu.org](http://www.gnu.org)  
(last updated 7 March 2024, for GNUUnited Nations version 1.5)

by Yavor Doganov <[yavor@gnu.org](mailto:yavor@gnu.org)>

---

This manual is a guide for the GNU Web Translators.  
Last updated on 7 March 2024, for GNUUnited Nations version 1.5.

Copyright © 2007, 2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020, 2021, 2022, 2023, 2024 Free Software Foundation, Inc.

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.”

# Table of Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Team Members</b>	<b>2</b>
2.1	Joining a Team	2
2.2	How to Submit a Translation	2
2.2.1	How to Submit a Translation in PO Format	3
2.2.2	How to Submit a Translation as Plain Text	3
2.3	Leaving a Team	4
<b>3</b>	<b>Team Co-ordinators</b>	<b>5</b>
3.1	How to Form a New Team	5
3.2	The Gentle Art of Managing a Translation Team	7
3.3	Peer Review	7
3.3.1	How to Track Tasks and Bugs Using Savannah	8
3.3.2	How to Proceed with Unreviewed Translations	9
3.4	CVS Commits and Best Practices	9
3.5	Taking Advantage of Savannah	10
3.5.1	Managing Members	11
3.5.2	Homepage of the Team	11
3.5.3	Support Tracker	11
3.5.4	Tasks Tracker	11
3.5.5	Bugs Tracker	12
3.5.6	News Tracker	12
3.5.7	Managing Mailing Lists	12
3.5.8	Version Control Systems	12
3.6	Promoting Members as Co-leaders	12
3.7	Reporting Team Status	13
3.8	How to Retire Painlessly	13
<b>4</b>	<b>Translation Process</b>	<b>14</b>
4.1	What to Translate	14
4.1.1	First priority	14
4.1.2	Second priority	15
4.1.3	Important Directories	15
4.1.4	Other Directories	15
4.1.5	Important Languages	16
4.2	Keeping Translations Current	16
4.3	Language-specific Terminology	17
4.4	When to CAPITALIZE	17
4.5	Fixing Bugs on Original Pages	17
4.6	How to Handle Internal Links	18
4.7	Distribution Terms	18
4.8	Copyright Notices	18
4.9	Editing PO Files	19
4.10	Related Mailing Lists	20
4.11	Savannah Project Membership	21
4.12	Summary of SSI #includes	21

4.13	Technical Pages.....	23
4.14	How to Use Custom CSS.....	23
4.14.1	Specific Issues Related to RTL.....	24
4.15	Migration to the New Style.....	24
<b>Appendix A GNU Free Documentation License.....</b>		<b>25</b>
<b>Index.....</b>		<b>32</b>

# 1 Introduction

This manual is an attempt to describe in detail the process of translating [www.gnu.org](http://www.gnu.org) articles—how to join a team, or start a new one, the responsibilities of the team members and leaders, as well as some peculiarities of the GNU Project’s website when it comes to localization.

The GNU website contains hundreds of documents, most of them philosophical articles (essays) and technical documents which need to be translated to make them available to a broader audience. This is especially important for the philosophy-related materials, as many people do not speak English and even those that do usually prefer to read such articles in their native language. Dealing with the task of translating a website this large is a hard job, and too often people volunteering as translators get frustrated or lose interest in keeping up with that work. Reading this manual, and the related GNUN manual (see Section “GNUnited Nations” in *The GNUnited Nations Manual*), is just the tip of the iceberg. This is not meant to discourage any potential volunteer; rather, we prefer to be honest and to give preliminary estimation of the work/responsibility involved—if you feel you are not in a position to help you may move on to a smaller project before going through all procedures.

It is important to realize that being a GNU Web Translator is a hard job at all levels, but your help is much appreciated and is invaluable contribution to the society. While there are many people who contribute to our community by writing free software (and their number is constantly increasing), the ones actively engaged in teaching others to appreciate and defend their freedom are only a few. Consequently and rather unfortunately, there are not so many volunteers willing to maintain in the long term translations of the various essays that describe the fundamental values of the free software movement.

Translators of the <https://www.gnu.org> website are organized in language teams. Each team has one or more co-ordinators, who are responsible for the respective team; they are also referred to as leaders or (when multiple in a single team) co-leaders. The co-ordinators participate in the Savannah ‘`trans-coord`’ organizational project, which is managed by the GNU Web Translation Managers (also known as Translation Managers or web-translators). The manual is organized in chapters that follow the organizational structure of the whole translation project.

For the issues common for all translators, see Chapter 4 [Translation Process], page 14. The sections of that chapter are sorted so that those interesting for less involved people (like occasional contributors) come first; the technical details tend to be at the end.

If you wish to join a translation team or contribute a translation or two, see Chapter 2 [Members], page 2. If your intention is to form a translation team, see Chapter 3 [Leaders], page 5.

## 2 Team Members

Being a team member means to co-operate with a group of other people, working under the co-ordinatorship of the appointed team leader. Usually, this involves translating articles and reviewing/proof-reading other people’s translations, participating in discussions about terminology issues, and sometimes performing clean-up tasks.

### 2.1 Joining a Team

To join a team, please first look at the existing teams in Translations README (<https://www.gnu.org/server/standards/README.translations.html#TranslationsUnderway>). Chances are that there is already an established team. If there is no team listed for your language, this means that:

- There is no team established and there are no translations to this language.
- Some translations were submitted by occasional contributors, but no team has ever been formed.
- The page is not updated to reflect the current situation (this shouldn’t happen, but it’s a possibility anyway).

If the team is marked as *orphaned* (“New coordinator needed”), there is no problem: you can still submit your translation to [web-translators@gnu.org](mailto:web-translators@gnu.org) (see Section 2.2 [Submitting], page 2). In case you want to establish a new translation team or become a co-ordinator of an existing one, please refer to the next chapter, see Chapter 3 [Leaders], page 5.

Contacting the team is best done via Savannah—each translation team has its own project, named ‘*www-lang*’, with the project page being ‘<https://savannah.gnu.org/projects/www-lang>’. All teams should have mailing lists, typically in the form [www-lang-...@gnu.org](mailto:www-lang-...@gnu.org). Some teams have homepages, ‘<https://www.gnu.org/server/standards/translations/lang>’ with additional contact details and procedures for team members.

You could also write directly to the team leader via the Savannah interface—that way your request will be recorded by Savannah and can be tracked or completed when the membership is approved.

The actual process of submitting translations for review varies from team to team, as teams have certain liberties to organize themselves as they see fit. Thus, this manual does not make any attempt to cover that aspect—please refer to the team-specific documentation (if any) or ask the co-ordinator.

Certainly, it is not mandatory to be an active team member to contribute a translation or two. If you feel that you don’t have the time to participate actively, that is fine; you can still send your translation to the team. No contributions should be rejected.

If you do not hear from the team within a reasonable time frame (say, two weeks), please write to [web-translators@gnu.org](mailto:web-translators@gnu.org).

For general information about the translation process, see Chapter 4 [Translation Process], page 14.

### 2.2 How to Submit a Translation

Everyone can still submit translations even if there is no translation team formed. There are two ways to do that—following the existing procedures, which is the preferred way, and sending it as plain text, which means more work for a limited group of volunteers (the Translation Managers) to convert the translation in `.po` format.

To make the work with PO files easier, team co-ordinators can write a team-specific guide for people who are not familiar with that format, like <https://www.gnu.org/server/standards/translations/po-how-to.html>.

### 2.2.1 How to Submit a Translation in PO Format

All translations<sup>1</sup> are maintained via GNUN (<https://www.gnu.org/software/gnun/>), which significantly eases maintenance and avoids the unpleasant situation where a translation is lagging behind the original. See Section “Advantages” in *The GNUUnited Nations Manual*.

Since September 2008 all new translations at gnu.org are installed in .po format, and the .html is generated automatically. Here are the steps to produce and submit such a translation:

- Make a checkout of the CVS Web repository of the ‘www’ Savannah project. You can find generic instructions at <https://savannah.gnu.org/cvs/?group=www>. All updates to the website are done as commits in the repository, so you would need an up-to-date working copy. Anonymous access works under any circumstances, i.e. it is not mandatory to have a registered account at Savannah to use it. You can also check out only a specific directory, for example:

```
cvs -z3 -d:pserver:anonymous@cvs.savannah.gnu.org:/web/www co www/gnu
```

This command will fetch only the /gnu directory—in other words, all articles at <https://www.gnu.org/gnu>.

You can also fetch single files by their respective URLs. For example, the URL for the template file of <https://www.gnu.org/philosophy/free-sw.html> is <https://www.gnu.org/philosophy/po/free-sw.pot>; the URL for its lang PO file (when available) is always <https://www.gnu.org/philosophy/po/free-sw.lang.po>:

```
wget https://www.gnu.org/philosophy/po/free-sw.pot
```

- Assuming you already know the article you want to translate, you have to create an empty `article.lang.po` file and then translate all messages with a PO editor. See Section “New Translation” in *The GNUUnited Nations Manual*. For an almost complete list of PO editors, see Section 4.9 [PO Editors], page 19.
- When you are pleased with the translation, check that the PO file is valid and submit it to [web-translators@gnu.org](mailto:web-translators@gnu.org), attached to your message. The web-translators will review (to the best of their ability) the translation and will install it in the repository.
- In order for GNUN to be able to generate (and subsequently update) a fully translated page, the language should have the server templates available as PO files. These templates are short, and translating them shouldn’t take much time. If the language code is present in the `TEMPLATE_LINGUAS` variable at `server/gnun/gnun.mk`, then you don’t have to do anything. If it is not, the required files are defined via the `extra-templates` variable in `server/gnun/gnun.mk`—you can translate and submit them in the usual way, together with the translation of the essay.

If you don’t want to translate the templates for whatever reason—do not worry, the web-translators will install empty templates (which means the English strings will be used).

It is quite possible that there will be errors or typos, so once you are informed that the translation is online, check it carefully and if necessary, resubmit the PO file with corrections. Do not forget to run `cvs update` first and edit the updated .po file from the repository—most probably the Translation Managers have already made some modifications to it, usually to fix validation errors and to complete the PO file header.

### 2.2.2 How to Submit a Translation as Plain Text

If you feel the procedure described in the previous section is too burdensome and unfeasible for you to follow, you can still submit a translation in plain text. It will be manually converted to PO file by the GNU Web Translation Managers, which can be tricky sometimes, and naturally, means more work for them and slower processing of your request.

---

<sup>1</sup> Well—not really, but the goal is to maintain all of them.

You should *never* translate the HTML markup—i.e. *do not* use the “View Source” functionality of your browser to translate the raw HTML. Most of it is irrelevant, and automatically inherited from the markup of the original article. Simply save your translation in a plain text file (`.txt`), preferably in UTF-8 encoding. You can use any decent text editor for that—Emacs, Vim, gEdit, Kate, LibreOffice (the file should be saved as `.txt`, not `.odt`), etc.

Translate the title, the main heading and the body of the article up to the footer. For example, for the Free Software Definition (<https://www.gnu.org/philosophy/free-sw.html>) that would be:

```
What is free software? - GNU Project - Free Software Foundation

What is free software?

The Free Software Definition

The free software definition presents...
...
...You can review the complete list of changes to the page through
the cvsweb interface.
```

Since web-translators do not speak all languages, it is essential to mark somehow any inner markup, because very often it is hard to figure out what translated text should be enclosed inside `<em>` or `<a>` elements, to name a few. The easiest way to do this is just to use the corresponding HTML markup, although anything else is suitable. For example, here is how to indicate that the link “History section” at the first paragraph of the same article should correspond to “sección historial” in the translation:

```
Si quisiera revisar los cambios que hemos hecho, por favor vea la
{sección historial} más abajo para más información.
```

It is not necessary to include the value of the `href` attribute, as it is already known.

If you wish your name to appear in the footer as a translator of the article, please also provide a translation of ‘`Translation: your name`’ or ‘`Translated by: your name`’, as you prefer. Please also state if you wish your email address to be published (some readers prefer to send suggestions directly to the translator, but we certainly do not require that translators must publish their address).

Finally, send the translation to `web-translators@gnu.org`, either inline or as an attachment to your message. Once online, please check for any errors or omissions that may have resulted from the conversion process, and report them back.

## 2.3 Leaving a Team

When you realize that you don’t have time or can’t devote sufficient resources to perform the tasks anymore, it is prudent to inform the translation team co-ordinator and possibly all the rest of the team-mates. The team leader should always have a rough estimation about the available translators, even though there are no reliable means to establish that. Your announcement that you are stepping down (temporarily or permanently) may help her in this regard.



## 3 Team Co-ordinators

A gnu.org translation team leader is the person who is ultimately responsible for organizing and managing the team, including, but not limited to, having the final say on contributed translations and exercising levels of control as she sees fit.

A prospective team co-ordinator should have perfect understanding of the GNU Philosophy and the various issues the free software movement set out to solve. Energy and time are always needed, as well as certain communication skills.

However, a team leader is not a dictator (for life); every action and decision taken should have its justification and should stem from the goals of the project at large. Inefficient or inoperative leaders are replaced, if necessary.

### 3.1 How to Form a New Team

Establishing a new team is not hard, but a certain procedure ought to be followed. The most important thing to realize is that this is somewhat a long-term engagement that requires a lot of spare time, communication and technical skills, and devotion. The only “bonus” team leaders have is more work and more responsibilities.

You should read *all* the documentation related to the translation process and at the very least all important philosophy-related articles listed on the Translation Priorities (<https://gnu.org/server/standards/translations/priorities.html>) page before you decide to form a new team, or take over an orphaned team. Once you have the internal feeling that having a gnu.org translation team for your language is a must, and you are the one for this job, follow these steps:

1. If you do not have a Savannah account, register at <https://savannah.gnu.org/account/register.php>. Write access to the repository and project membership is handled via Savannah, so you would need an account in any case.
2. Checkout a complete working copy of the CVS Web repository as described at <https://savannah.gnu.org/cvs/?group=www>. If you are not yet member of any Savannah project, refer to the instructions under “Anonymous CVS Access”. If you are already a member of (any) Savannah project, you can proceed with “Project Member CVS Access via SSH”, although you will still lack permission to commit (later, when it is granted, you can use the same working copy).

Examine the layout and structure of the repository. Basically, it is mapped to the URL locations, more or less. Take a look at the most important materials to translate under `/philosophy`, `/gnu`, `/distros`, `/education` and `/licenses` directories just to get a rough estimate about the amount of work involved<sup>1</sup>. If you are still not scared and determined to go on further, excellent.

As you have probably observed, every directory that contains translatable articles has a `/po` sub-directory, which is where the canonical source format of the translations is stored.

3. Submit your first message stating that you would like to establish a new team to [web-translators@gnu.org](mailto:web-translators@gnu.org); please mention that you have read all the documentation and list the issues that remain unclear for you. The Translation Managers will answer your questions and send you the standard questionnaire for new team leaders. It is short and shouldn't take more than 10–30 minutes to complete. This questionnaire is important, as we consider it crucial for any translation team co-ordinator to have a good understanding of the philosophy of the free software movement.

---

<sup>1</sup> As of January 2021, there are over 300 files to translate in “important” directories; their volume is about 5 MB.

4. Check if your language code is present in the variable `TEMPLATE_LINGUAS` in the file `server/gnun/gnun.mk`. If it is not, the first thing to do is to translate and submit to `web-translators@gnu.org` the following files (all in the `server/po/` directory):
  - `head-include-2.lang.po`
  - `body-include-1.lang.po`
  - `body-include-2.lang.po`
  - `bottom-notes.lang.po`
  - `footer-text.lang.po`
  - `outdated.lang.po`
  - `top-addendum.lang.po`

See Section “New Translation” in *The GNUUnited Nations Manual*.

- The language code (*lang*) should be the ISO 639-1 code of the language, for example ‘hy’ for Armenian or ‘el’ for Greek. If the language is a variant such as Brazilian Portuguese or Simplified Chinese, use small caps and a dash—‘pt-br’ and ‘zh-cn’ instead of ‘pt\_BR’ and ‘zh\_CN’.
  - The PO file header and initial comments should be filled as documented.
5. Any prospective team leader should submit a few translations first. This is a process of pointing errors and omissions (which are expected and natural); it’s an important thing to do as the leader is going to carry out these checks on her own, once the team is approved. If there are existing translations that are not yet in PO format, the best thing to do is to migrate one or two. You can use `find` to find out what’s already in the repository, for example:

```
find -name \*.lang.html
```

6. Submit at least two translations of your own. We maintain a list with priority articles on the Translation Priorities (<https://www.gnu.org/server/standards/translations/priorities.html>) page, although it is probably hard to start with one of them. Choose whatever you wish, provided it is an essay and not an auxiliary page. Avoid translating the homepage or `planetfeeds.html`—they are moving targets and keeping up would be only a distraction for both parties in the process. As usual, send the completed translation to `web-translators@gnu.org`.
7. The Translation Managers will review your translations, and eventually comment on them (mostly technical details if there is no one among them speaking your language). Depending on the case, it might be required to submit a corrected file. In any event, please take into account the remarks in future work.
8. If all goes well, you will receive a response inviting you to apply for a new translation project at Savannah. The project name should be ‘`www-lang`’ where *lang* is, unsurprisingly, the language code. If such a project already exists, this step will be skipped and you’ll be made an administrator of the project and its mailing lists. To register the project, go to <https://savannah.gnu.org/register/> and make sure you fill in the required fields. The “Group type” should be ‘`www.gnu.org translation team`’, and “Project license”—‘`WebSite Only`’. In the “Tarball URL” field enter a bogus URL such as ‘<https://www.gnu.org>’.

**Pay attention:** This step is a formality. You should proceed with the project registration only when you have been asked by `web-translators@gnu.org` to do so. Otherwise, the submission may appear in the task list of the Savannah Hackers for a fairly long time, which is troublesome.

9. When the project is approved, the team information will be added to the list at `README.translations.html`, you will become a member of the ‘`www`’ project (thus

granting you CVS write access to the whole repository—so be careful) and the ‘**trans-coord**’ project. You’ll also be subscribed to the following mailing lists:

- `www-commits`
- `trans-coord-discuss`
- `www-discuss`

You’ll also receive monthly automatic reports about outdated translations. Please contact the Translation Managers if you’d like to receive them at a different email address.

10. When you are appointed the admin of the new project, please edit its configuration; in particular, write its description, create a mailing list (don’t forget to subscribe yourself!), optionally add a home page using Web CVS repository.

If you are taking over an orphaned team, the Translation Managers will make you the owner of its mailing lists (if any).

The whole process should not take more than two weeks or maximum a month—if this period turns out to be longer, it is an indication that you do not have the required time and resources for this job, or web-translators are badly lagging behind and do not process the requests with the expected pace.

In general, we try to avoid processing applications for new teams in parallel and direct all new volunteers to the person who is already establishing the team—this is also a verification if the prospective leader can co-operate easily with others.

The procedure for taking over an orphaned team is the same. Once completed, you will be made an admin of the respective ‘**www-lang**’ Savannah project, or if it doesn’t exist, invited to apply for registration. Do not automatically remove old members just because you are starting “afresh”—some of them might want to continue to contribute. Contact them privately, explaining that you’re the new appointed team co-ordinator, and ask them if they would be willing to continue their involvement in the team.

## 3.2 The Gentle Art of Managing a Translation Team

It is not our ambition to describe all activities involved in managing a team—it’s very likely that you will encounter new problems, take care of tasks nobody else is aware of, or invent new techniques and approaches in your quest to keep things running. Managing a team is a hard task on all counts: communication with others, recruiting volunteers (and keeping them as long as possible), defending certain decisions, leading discussions about terminology issues, handling personal conflicts within the team, technical skills when reviewing/merging/syncing translations, etc. The list goes on and on.

This manual can only summarize some of the most common issues and *suggest* ways to deal with them. It is up to the team leader to establish the precise team procedures and practices.

The `trans-coord-discuss@gnu.org` mailing list was specifically created to discuss issues that leaders encounter while managing the teams, and for general organizational work. Feel free to discuss anything related to the translation process there.

It is strongly recommended that translation teams attempt to recruit native English speakers in order to improve their translation process. Translators sometimes misunderstand English idioms and expressions, and as a result, they translate them incorrectly or in ways that are suboptimal and confusing. These errors are trivial to discover for the native English speaker.

## 3.3 Peer Review

First and foremost, find at least one person for peer review. You will review her translations, and she will review yours (at least in the beginning). Being a team leader does not mean that

you cannot make mistakes; everyone does. The mutual review (especially if done by a larger group) is crucial for the quality of the translation process. Too many errors are just missed (especially if they are obvious) when the translator does a final review of her own translation.

It is good to establish a practice: Do not commit officially (i.e. in ‘`www`’, which will appear online at <https://www.gnu.org> immediately) a translation that is not yet reviewed by someone else who is not the translator. Always perform a final review yourself even if the translation has been checked by another member of the team. In other words, every translation installed at `gnu.org` should pass through your hands (read: eyes).

One common technique for performing such reviews is to use a mailing list—the translator sends the new translation and participants comment on specific parts, quoting them appropriately. The benefit of this approach is that it is straightforward, but the drawback is that there is no automatic “record” about the conclusion of the specific discussion (or sub-thread) and sometimes such discussions easily digress, making it even harder to come up with a solution.

Another way is to use Savannah’s built-in trackers (the ‘**Tasks**’ and ‘**Bugs**’ trackers, specifically). This is further explained in the next section, see Section 3.3.1 [Tracking Tasks], page 8. One way or another, you should create some kind of review process.

### 3.3.1 How to Track Tasks and Bugs Using Savannah

The team leader has to make sure that prospective translations are reviewed, that they do not contain obvious errors and confusing expressions and that they match the spirit and intention of the original essay. However, many teams tend to suffer from a specific problem: team members rely on the leader to make these extensive reviews. That is fine, as far as it goes, and the leader should always review translations before installing them in the repository—but it is nearly impossible (especially for a large team) to rely on a single person for such tasks. Team co-ordinators often do not manage to make such reviews in time, resulting in frustration among the team members and generally slowing the translation process.

A solution to this specific problem is to distribute the load among more people. For example: Member D makes a translation of `foo.html` and uploads `foo.lang.po` in the translation project’s repository at Savannah, marking the relevant task as “Ready For Test” (of course, the equivalent is sending a message with the attached translation to the team’s mailing list, or similar). Then Member A, B and C (or only A and B if C is currently busy) review it independently and post comments/suggestions/errors in the bug tracker. Discussion goes on between them and D, problems are rectified and finally the leader (who may happen to be one of A, B, C, D) makes a final review. It is easier to make the final review when most of the issues are already fixed in previous revisions. Finally, the translation is published. The result is better quality of the translation (since more people looked at it) and the whole burden does not fall solely on the shoulders of the leader. You can also set up an internal formal rule: If a member makes a translation, he has to review another one (or two) as well.

Some translations can take a fairly long time—the typical example is a complicated essay or a transcript of a speech. It is best to avoid duplicate work by indicating, or better—recording, that someone is working on this specific article. The ‘**Tasks**’ tracker is suitable for this purpose.

It is prudent to discuss the most convenient naming scheme and practice among team members, and publish the convention or rules at the team’s homepage. Note that you can create *Custom Fields* in the trackers, and resolved bugs can be searched based on these custom values.

Thus, a possible straightforward way to manage these tasks is:

- If someone starts working on a new translation, she creates a new task with a ‘**Subject**’ indicating the article, for example simply ‘`philosophy/bsd.html`’ and assigns it to herself.
- When the translation is finished and ready for review, the translator changes the ‘**Status**’ to “Ready For Test”.

- Other members review it, and open bugs relevant to one specific problem. It is usually better not to conflate two different issues together—it makes them harder to discuss, and hard to track them by severity. Some are grammatical errors, some are fundamental ones that change the whole meaning, some are simply suggestions for improvement. It helps if the project admin creates new Category fields for every article, for instance ‘gnu/gnu-history.lang.html’, ‘philosophy/microsoft.lang.html’—it would enable functionality like “Show me all bugs ever reported against this translation”, which is useful.
- Once the bugs (or at least the important bugs) are fixed, the team leader can make the final review and install the translation in the official repository, marking the task as ‘Done’. Bugs that are not resolved should remain open, naturally.

If there are compelling reasons, teams can choose to manage these things using external resources and eventually other bug (or issue) tracking systems. Whatever you decide, please make sure that bugs can be reported using free software only, and that the software providing that service is free. It makes an extremely bad impression if a reader has to report a problem about a gnu.org translation via nonfree hosting platforms like SourceForge.

If you use a certain facility (i.e. a bug tracking system) to manage bugs in translations, it is best to take advantage of `generic.lang.html` and advertise it on every page. See [generic.html], page 22, for details.

### 3.3.2 How to Proceed with Unreviewed Translations

Sometimes a translation (typically your own) is not reviewed by anyone else for a fairly long time. This is unfortunate, but there is no reason to keep it in draft state forever. If nobody reviewed it for a substantially long period (like 3 or 4 months), review it yourself and commit it as it is. Readers may report bugs as well (and they do!).

It is important to record somehow that this published translation still lacks appropriate review. If the suggestion in the previous section is implemented, it would mean leaving the relevant task ‘Open’ and ‘Ready For Test’ despite the translation being officially online. You may also add a comment to the PO file.

## 3.4 CVS Commits and Best Practices

As all team leaders have write access to the CVS repository of the ‘www’ project, this technically means that they are able to modify every single file in it. This vote of confidence should never be abused—the only files team co-ordinators should add/update are those relevant to their translation work. It is OK to fix an obvious typo in an original article; for anything else please report to [webmasters@gnu.org](mailto:webmasters@gnu.org).

If you wish to volunteer as webmaster and help with generic webmaster work and RT tickets, that is perfectly fine—please follow the established (by the GNU Webmasters) procedure. If you are approved, you can modify such pages wearing your “webmaster’s hat”.

If a particular page has issues with the markup which create problems for your language, please inform [trans-coord-discuss@gnu.org](mailto:trans-coord-discuss@gnu.org). For general issues that affect more articles, or for severe problems, please write to [www-discuss@gnu.org](mailto:www-discuss@gnu.org).

If you are not familiar with CVS, it is recommended to read CVS manual, for a basic understanding of how this VCS works. See *Version Management with CVS*. It is not necessary to become an expert—the ‘www’ project does not use complex features like tags, vendor branches, merging, etc. as they are not very useful for a live website.

However, you’d probably have to learn how to use CVS for effective work—to extract information from the history, review diffs and specific changes, synchronize with the working repository of the team (if any), adding/removing files, etc.

If you make changes that affect more than one file but the change is coherent, please do it as a single commit. This will generate only one message to `www-commits@gnu.org`, which is better than 5 messages for 5 files about semantically the same change. Always write commit logs in English<sup>2</sup>, providing a short description of the change. If you modify a file that is not an article but a script or part of software (such as `server/gnun/gnun.mk`), it would be nice to follow the GNU Coding Standards and describe the change precisely (see Section “Change Logs” in *The GNU Coding Standards*). For example, do not write:

```
Added support for Nepali.
```

or

```
Yay! First commit of the Panjabi homepage!
```

Instead, write the log as follows:

```
(TEMPLATE_LINGUAS): Add 'ne'.
```

and

```
(FUZZY_DIFF_LINGUAS): Add 'pa'.
```

This makes it easier for others to search for a particular change in the history.

If you add a binary file (for example, `.png`), do it with `cvs commit -kb file`. This turns off keyword substitution, which prevents RCS keywords like `$Id$` to get expanded, subsequently corrupting the file. See Section “Substitution modes” in *Version Management with CVS*. More importantly, using `-kb` prevents corruption of the binary when people using CVS clients under infamous OS checkout modify the file, and then commit it with messed ends of lines.<sup>3</sup>

Although not absolutely compulsory, it is recommended that every team leader subscribes to `www-commits@gnu.org`. It is useful to examine the diffs of your own messages, if you miss something while inspecting the diff before the commit. In any case, a team leader should be subscribed to that list to avoid his own commit messages to be moderated. If you absolutely do not desire receiving all traffic, just disable mail delivery in Mailman’s user interface.

### 3.5 Taking Advantage of Savannah

Every translation team should have a project in Savannah. There are some teams that use their own resources outside Savannah; although there’s no obligation to use Savannah for team work, the need for a Savannah project for each language is obvious: it’s a standard way to find information for translation teams and their contacts.

Using external hosting facilities may seem justified sometimes. Some teams may have already established repositories or bug tracking systems where usual contributors already have access. Some team members prefer to work within the established infrastructure of a broad translation team (for whatever reason), but this is discouraged. It is required that every team has a mailing list at Savannah (see Section 3.5.7 [Savannah Mailing Lists], page 12), because it is easier to pass its management to the new co-ordinator when the old one steps down, and it helps to keep the archives at one place for future members of the team. Likewise, it is better to use Savannah for team’s repository and bugs/tasks.

However, it is important to remember that regardless of the technical resources which a team decides to use, the responsibility of the team co-ordinator remains the same.

Those teams that are using Savannah have a broad variety of tools at hand: team membership management, documents, trackers (bugs, tasks and support), alerts, CVS (and any other VCS that Savannah supports), home pages, etc. How each team uses these resources is up to the

---

<sup>2</sup> This advice is applicable for the ‘`www`’ repository only—feel free to write logs in your native language when committing in your team’s repositories.

<sup>3</sup> Few years ago there still were committers using nonfree operating systems—we don’t dictate what OS people use, but we can at least prevent this technical kind of damage.

team itself, but it often turns out to choose Savannah for nearly all of the team activities, as it requires almost zero work; the Savannah Hackers are happy to support us.

Whatever you (in your capacity as a team leader) decide, please do it with caution: some organizational decisions may become ineffective as time goes by, and some may not scale well when the team grows. If the team is young and has a couple of members, it is better to refrain from such decision and discuss them with all the members when their number grows. Two or three people do not need a rocket platform or complex wizardry to do their work.

The next sections contain suggestions about how a team can use the facilities provided by Savannah. It is not mandatory to follow them, they are just suggestions.

### 3.5.1 Managing Members

You should add active translators as members of the translation team, and remove them when they leave. Team members should have access to all of the project's resources, and tracking their number is one of the ways for web-translators to determine the status of the team.

It is OK if a particular contributor wants to translate an article or two and does not want to be engaged with the team on a long-term basis. In such situations, there is no need to add her as a member.

It is a good idea to mark inactive members, for example if there is no interaction (bug reports, new translations, updates to existing translations, proof-reading) for at least six months. You can do that by unmarking the 'On Duty' checkbox for the respective project member under 'Set Permissions'. Inactive members have absolutely the same rights as active ones—the only exception is that they don't count for the total number of members, and they appear separately on 'View Members'.

### 3.5.2 Homepage of the Team

Every Savannah project has a Web repository, which is, for technical and historical reasons, only CVS. By default it is mapped to '<https://www.gnu.org/server/standards/translations/lang>'; to add files to it first make a checkout, following the instructions at '<https://savannah.gnu.org/cvs/?group=www-lang>'.

It is recommended to describe all team-specific procedures, if there are any. That way, you can point potential team members to the corresponding page containing these instructions, instead of repeatedly explaining every volunteer separately.

All team-specific pages should follow the usual linking criteria in GNU Webmastering Guidelines (<https://www.gnu.org/server/standards/README.webmastering.html#pollinking>), and the FSF HTML Style Sheet Guidelines (<https://www.gnu.org/server/fsf-html-style-sheet.html>).

### 3.5.3 Support Tracker

This tracker is supposed to be related to things about the *project management* itself, i.e. project members may report here missing functionality and features that requires the project admin's action. Do not use it for anything else as it quickly becomes confusing. It is OK to disable it if the team is small.

### 3.5.4 Tasks Tracker

This is a way to manage all sorts of tasks. They appear in the personal Savannah page of the assignee, so it is difficult to miss them out. It is possible to use this tracker to “announce” to the team members that a specific article should be translated. The one who volunteers may assign the task to herself.

Teams may use this tracker to avoid duplicate work, by declaring that they intend to work on a specific translation.

Feel free to organize the ‘**Tasks**’ management as you see fit.

### 3.5.5 Bugs Tracker

The ‘**Bugs**’ tracker is designed for tracking bugs. You can use for several purposes:

- Suggest readers to report bugs there.
- Use it for all kinds of internal team tasks.
- Forward bugs reported against *lang* translations in the ‘**trans-coord**’ project and assign them to the specific maintainer (who is supposed to be a ‘**www-lang**’ project member), if you have such policy.

### 3.5.6 News Tracker

That is a way to inform newcomers and interested people (who visit the project page from time to time, or subscribe to the ‘**News**’ RSS feed) about a major change or event within the project.

You can also setup news entries to be sent to a mailing list (that’s possible for the other trackers as well).

The purpose of this feature is informational—if members need to know about an important change (in practices, procedures, etc.), it is perfectly OK to announce it here. Some teams use it to announce new translations, which is also fine.

### 3.5.7 Managing Mailing Lists

Every team should have a mailing list on lists.gnu.org and use it for internal communications. All active translators should be on the list. The list owner should be the co-ordinator of the team. The name of the list should begin with ‘**www-lang-**’. The team co-ordinator is in the position to decide about the settings like being public or private.

The list will make it possible for the GNU project to contact the team when the co-ordinator disappears; its archive will also give access to the history for new translators.

You can create new mailing lists via the Savannah interface. However, this should be done after some thought. If the project membership is low ( $\leq 10$  members), there is no need to create more than one mailing list.

You can redirect all messages generated by the trackers to any list.

### 3.5.8 Version Control Systems

An easy way to keep up with changes in the original articles and to manage continuous contributions is to keep all translations in the translation project’s Sources repository. That way, it is easy to edit draft translations and install them in ‘**www**’ only when they’re ready. It is also convenient to update the translation (merge any changes from the original) while it is still under review.

See Section “Team’s Repository” in *The GNU United Nations Manual*, for more information.

**Remember:** A choice of a particular VCS is a sensitive matter—some modern ones provide compelling features, but they also bump the barrier for participation higher. The VCS is supposed to ease collaborative maintenance—if it eases only you, project members just won’t use it so that won’t be a net win.

## 3.6 Promoting Members as Co-leaders

When the team grows large and it becomes hard for a single person to manage, there is no problem to add another (or even two other) people to help. Note that a subsequently appointed team co-ordinator is not simply a *committer* with write access to the ‘**www**’ repository; she has full responsibilities just like a single leader, although the latter still remains the primary contact for the team.



If you'd like another person to act as a co-leader and help you with the management tasks, send a message to `web-translators@gnu.org` with her name and Savannah account. She has to be already an administrator of `'www-lang'`.

The procedure for co-leaders is a simplified version of the one for a new team or taking over an existing team. See Section 3.1 [New Team], page 5.

To remove co-ordinators, please write to `web-translators@gnu.org` with details and rationale for the removal. Do not edit `README.translations.html` yourself; this is a final formality step to be performed by the Translation Managers.

## 3.7 Reporting Team Status

Team leaders must send an annual report about the status of the team. A good report should include:

- General information about the team's accomplishments during the past year, like:
  - A list of new translations.
  - New members since the last report.
  - Solved problems and other issues, if any. (Usual bug reports and other improvements/fixes to the existing translations do not count as *problems* in this sense.)
- Current active members.
- Current problems (technical or social), conflicts, and ideas for sorting them out.
- Anything else you consider important or worth mentioning.

The best time to send a report is near the end of the year, for example November.

If there is no sensitive information in the report and you feel like sharing it, you can send it to `trans-coord-discuss@gnu.org` (which is still a private mailing list). That way, other list readers may help with suggestions how to solve a particular issue. Informing each other about the progress improves the community spirit.

If you do not wish to share some information that is in the report, please send it to `web-translators@gnu.org`.

## 3.8 How to Retire Painlessly

When you feel you don't have the energy to manage the team successfully, or perhaps you start losing motivation, please inform `web-translators@gnu.org`. It would be substantially easier if you try to find a replacement or recommend a specific person—we will try to find someone in any case, but your judgment is important and it will be considered with priority.

An excellent way to step down is to do it with a “plan”—suggest the person you consider capable of doing the job as co-leader (see Section 3.6 [Co-leaders], page 12) and retire completely when she is absolutely ready to proceed without your further help and advice.

## 4 Translation Process

In general, it is expected that all participants in the translation process apply common sense for all of the decisions (important or not) they are going to take in their capacity as a manager, team leader, or contributing member. Certainly, many decisions are not easy, and require some thought.

This manual is a work in progress—it is not set in stone, and it will never be finished—the ultimate goal is to constantly improve the translation process, and as a consequence, the documentation. Every participant in the process should be free to suggest modifications to the current procedures and suggestions how to improve the current state of affairs. Ideally, they should be accompanied with patches to the Texinfo source, but that's not mandatory. In any event, please write to [trans-coord-discuss@gnu.org](mailto:trans-coord-discuss@gnu.org)—the goal of this list is precisely to discuss improvements of the translation process.

### 4.1 What to Translate

This section lists translation priorities. You can find links to automatic reports about current status of translations of all active teams sorted by their priority in GNUN Reports (<https://www.gnu.org/software/gnun/reports/reports.html>). If the page for your team is missing there, please ask [web-translators@gnu.org](mailto:web-translators@gnu.org) to add it to the cron job.

#### 4.1.1 First priority

This list of web pages from <https://www.gnu.org> that have been selected as translation priorities (in no particular order).

- The Free Software Definition (<https://www.gnu.org/philosophy/free-sw.html>)
- Free Software Is Even More Important Now (<https://www.gnu.org/philosophy/free-software-even-more-important.html>)
- How Much Surveillance Can Democracy Withstand? (<https://www.gnu.org/philosophy/surveillance-vs-democracy.html>)
- Why Software Should Not Have Owners (<https://www.gnu.org/philosophy/why-free.html>)
- Why Open Source misses the point of Free Software (<https://www.gnu.org/philosophy/open-source-misses-the-point.html>)
- Did You Say "Intellectual Property"? It's a Seductive Mirage (<https://www.gnu.org/philosophy/not-ipr.html>)
- Avoiding Ruinous Compromises (<https://www.gnu.org/philosophy/compromise.html>)
- Free Software and Education (<https://www.gnu.org/education/education.html>)
- Why Schools Should Exclusively Use Free Software (<https://www.gnu.org/education/edu-schools.html>)
- Vocational Higher Secondary School Irimpanam (<https://www.gnu.org/education/edu-cases-india-irimpanam.html>)
- Tux Paint (<https://www.gnu.org/education/edu-software-tuxpaint.html>)
- The GNU Project (<https://www.gnu.org/gnu/thegnuproject.html>)
- GNU/Linux Distros (<https://www.gnu.org/distros/distros.html>)
- Free System Distribution Guidelines (<https://www.gnu.org/distros/free-system-distribution-guidelines.html>)
- Explaining Why We Don't Endorse Other Systems (<https://www.gnu.org/distros/common-distros.html>)

- How to choose a license for your own work (<https://www.gnu.org/licenses/license-recommendations.html>)

### 4.1.2 Second priority

These pages are the second level of priority. Once the pages in the list above are done, please translate these next.

- What is Copyleft? (<https://www.gnu.org/licenses/copyleft.html>)
- Copyleft: Pragmatic Idealism (<https://www.gnu.org/philosophy/pragmatic.html>)
- Selling Free Software (<https://www.gnu.org/philosophy/selling.html>)
- Who does that server really serve? (<https://www.gnu.org/philosophy/who-does-that-server-really-serve.html>)
- The JavaScript Trap (<https://www.gnu.org/philosophy/javascript-trap.html>)
- Why Educational Institutions Should Use and Teach Free Software (<https://www.gnu.org/education/edu-why.html>)
- Escuela Cristiana Evangelica de Neuquen (ECEN) (<https://www.gnu.org/education/edu-cases-argentina-ecen.html>)
- Ambedkar Community Computing Center (AC3) (<https://www.gnu.org/education/edu-cases-india-ambedkar.html>)
- Educational Free Software (<https://www.gnu.org/education/edu-software.html>)
- GCompris (<https://www.gnu.org/education/edu-software-gcompris.html>)
- Educational Frequently Asked Questions (<https://www.gnu.org/education/edu-faq.html>)
- The Education Team (<https://www.gnu.org/education/edu-team.html>)
- Linux and the GNU Project (<https://www.gnu.org/gnu/linux-and-gnu.html>)
- The Right to Read (<https://www.gnu.org/philosophy/right-to-read.html>)
- Why Free Software needs Free Documentation (<https://www.gnu.org/philosophy/free-doc.html>)

### 4.1.3 Important Directories

Essays and articles in the following directories should be translated in all available languages:

- /education
- /gnu
- /licenses
- /philosophy
- /proprietary

In these important directories, however, there are articles of historical or peripheral interest only, for which new translations are not necessary. For instance, `/philosophy/sco/*`, `/philosophy/*-old.html`, `/licenses/*_seminar.html`, `/philosophy/economics_frank/frank.html`, etc. Please use your judgment. (The existing translations of these articles should be maintained, though.)

The `/proprietary` directory is important, but it changes as fast as a newsfeed. Don't start translating it unless you intend to update the translated pages over the long term.

### 4.1.4 Other Directories

Don't bother translating the pages in `/people`, except perhaps `speakers.html`. Experience shows that translations of the other pages in that directory require too much maintenance work, and

they don't provide much benefit. Translating `/thankgnus` may likewise be considered a waste of time.

There is no problem to translate `/home.html` if you have a very active team, but don't make the mistake to pick it up as your first translation. It is modified often, sometimes intensively, and only active team members should take that road.

The material in the `/software` directory pertains to individual GNU packages. If you would like to translate something in that directory, please talk with the maintainers of the package to see what they would like to do.

### 4.1.5 Important Languages

We welcome translations of essays into any language, but the following are particularly important:

- French
- Spanish
- Portuguese
- Chinese
- Arabic
- Indonesian
- Russian
- Japanese
- Hindi

## 4.2 Keeping Translations Current

It is very important to keep existing translations up-to-date with the respective English originals. This task should be higher priority than translating new articles. We developed various means to automate the process of tracking outdated translations.

- GNUN's `report` rule can help you identify precisely which articles need updating; see Section "report" in *The GNUUnited Nations Manual*. There is a monthly cron job which sends the output of this rule to each team as requested by their leaders. If you want the addresses changed, please write to `web-translators@gnu.org`.
- The `gnun-report` script produces a HTML page listing detailed status of translations; see Section "gnun-report" in *The GNUUnited Nations Manual*. A cron job commits updated reports for all active teams to GNUN project web repository, typically twice an hour. The links to those reports are provided on the GNUN Reports (<https://www.gnu.org/software/gnun/reports/reports.html>) page.
- GNUmakefile.team provides a more detailed `report` target: unlike the output of the previous tools, it analyzes the status of files in team's repository as well as of those in 'www' repository; see Section "report in GNUmakefile.team" in *The GNUUnited Nations Manual*.
- GNUmakefile.team also has a means to send more detailed reports to specific translators; see Section "notify in GNUmakefile.team" in *The GNUUnited Nations Manual*. The notification facility takes the output of the `report` target, adds the URLs of relevant files, and the results are sent with attached HTML files of team's-against-'www' differences to the translators who requested tracking particular files.

The feature is supposed to be invoked via a cron job; such jobs already run for some teams on our server. If you'd like GNU Web Translation Managers to setup a job for your team, please write to `web-translators@gnu.org`.

- If your editors don't highlight differences against previous messages, you may find it useful to track the changes in the messages with `gnun-add-fuzzy-diff`. For more details, see Section “gnun-add-fuzzy-diff” in *The GNUited Nations Manual*.

### 4.3 Language-specific Terminology

This is a very important topic, not yet covered by this manual.

Some tips are given in Translations README (<https://www.gnu.org/server/standards/README.translations.html>).

### 4.4 When to CAPITALIZE

The English language has some rules for capitalization of titles, chapters, acronym expansions and the like. These rules are neither strict nor uniform, although the gnu.org website strives to apply them consistently. They do not make sense for many other languages, but unfortunately, many translators *erroneously duplicate* the capitalization in their translation.

Examples for common (and correct) English capitalization is the title of the article “Why Software Should Be Free” or “Free Software Foundation” (FSF). However, in languages that do not have such grammar rules it is wrong to write “Dlaczego Oprogramowanie Powinno Być Wolne” (Polish) or “Fondation Pour Le Logiciel Libre” (French).

Another prominent and widely spread mistake is to write your own language with a capital letter in the list of translations when languages are written beginning with a small letter according to your own rules<sup>1</sup>. In other words, it is right to write ‘English’ or ‘Deutsch’ (because in English and German languages are capitalized), but not ‘Français’ or ‘Português’—write them as ‘français’ or ‘português’, respectively.

### 4.5 Fixing Bugs on Original Pages

GNU webmasters proofread the texts before posting them and fix the bugs reported by www.gnu.org visitors, but occasionally some mistakes do slip into the pages. Translators are probably the people who read the pages most carefully, so it's them who are likely to find those mistakes first of all.

There is a trend for translators to fix the bugs in their translation instead of the original page. This is understandable: you have to contact additional people (webmasters) in order to fix it on the English page, so it's easier just to make the translation. However, there are reasons why you shouldn't leave bugs on the English pages:

- The text may not be wrong, but you misunderstand it. (This, in turn, may mean that the text should be reworded to make it unambiguous.)
- It's as important to have the English version correct as your translation.
- Some mistakes may propagate to translations in other languages. Also, when the page changes, the translators have to update their translations, so the earlier you fix the bugs, the fewer people will have to update their translations.

The bugs are reported to [webmasters@gnu.org](mailto:webmasters@gnu.org); if you are a team leader, you can also discuss the issues on [www-discuss@gnu.org](mailto:www-discuss@gnu.org).

---

<sup>1</sup> The lists of translations are generated automatically. The names of the languages are defined in a specific file, `languages.txt` (see Section “languages.txt” in *The GNUited Nations Manual*).

## 4.6 How to Handle Internal Links

In short, you should leave the URLs in links to other articles of [www.gnu.org](http://www.gnu.org) as they appear in the English text.

These days [www.gnu.org](http://www.gnu.org) uses HTTP language negotiation to provide the most preferred translation available according to user's browser settings. The texts of articles use generic URLs like `‘/directory/article.html’` (note no language suffix). When the visitor follows such links, [www.gnu.org](http://www.gnu.org) chooses the best translated version, or the English version if there is no suitable translation.

Once upon a time, there was a practice to link to the respective translation (`‘/directory/article.lang.html’`) when available. You shouldn't do this any more. First, new translations are added, and occasionally even removed, and timely updating the links in all existing translations is not feasible. Second, and more important, visiting a translation doesn't really imply that its language is the most preferable one.

For instance, let us imagine that visitor's native language is Serbian, and she can also understand Bulgarian. Then (as of Sep 2015) the best version of Enforcing the GPL (<https://www.gnu.org/philosophy/enforcing-gpl.html>) is Bulgarian; however, that page links to the Free Software Definition (<https://www.gnu.org/philosophy/free-sw.html>), which is available both in Serbian and in Bulgarian. If the Bulgarian translation of the announcement linked to the Bulgarian version of the definition, the visitor would be directed to a wrong translation.

## 4.7 Distribution Terms

Most [www.gnu.org](http://www.gnu.org) articles are released under the terms of the Creative Commons Attribution-NoDerivs 3.0 United States or 4.0 International license. The exact HTML for English pages to use is:

```
This page is licensed under a <a rel="license"
href="http://creativecommons.org/licenses/by-nd/3.0/us/">Creative
Commons Attribution-NoDerivs 3.0 United States License</a>.
```

Pages in other languages should translate this notice, and should link to a translated version of the Creative Commons license “deed” if it's available.

The translation should be used consistently throughout pages translated to that language.

For example, here's the text they provide for Dutch:

```
Dit werk is gelicenseerd onder een <a rel="license"
href="http://creativecommons.org/licenses/by-nd/3.0/us/deed.nl">
Creative Commons Naamsvermelding-GeenAfgeleideWerken 3.0 Verenigde
Staten licentie</a>
```

Note that the link in this text is changed to point directly to the Dutch language deed. We should always link to a copy of the license deed that's in the same language as the page itself. When no deed in your language is available, link to the English language deed.

Note that translations should *not* change the jurisdiction of the license; in case of CC BY-ND 3.0, they should *always* link to the CC BY-ND 3.0 *United States* license, and *not* a different port like CC BY-ND 3.0 Japan. This is because there are substantive differences between the way different ports handle moral rights issues, and we prefer the specific terms that are in the United States license.

## 4.8 Copyright Notices

When translating an article, the translator makes a derivative work, and the copyright belongs to the translator unless disclaimed, assigned, or the translation is made for hire.

The translation should include copyright notices for the original work and for the translation. The copyright notice for the original work is copied from it, the copyright notice for the translation should include a note making it clear that it applies to the translation, and list the years when the translation had copyrightable changes.

Let us assume, for example, that Richard Stallman wrote an article in 1998 and updated it in 2005, 2008, 2015, and 2017; its copyright notice looks like,

```
Copyright &copy; 1998, 2005, 2008, 2015, 2017 Richard Stallman
```

Then, Besnik Blea translated it in 2013 and updated in 2016 and 2017. The copyright notices in the translation should look like,

```
Copyright &copy; 1998, 2005, 2008, 2015, 2017 Richard Stallman
Copyright &copy; 2013, 2016, 2017 Besnik Blea (translation)
```

When the copyright holders for the translation and the original article coincide, the translation should include a single copyright notice listing the copyrightable years.

For example, the copyright was assigned to the FSF both by the author of the English article and by the translators. The article was written in 2013 and updated in 2018. The translation was made in 2015 and updated in 2018. Then the copyright notice in the original article looks like

```
Copyright &copy; 2013, 2018 Free Software Foundation, Inc.
```

The copyright notice in the translation should look like,

```
Copyright &copy; 2013, 2015, 2018 Free Software Foundation, Inc.
```

In all cases, all parts of the copyright notice should be in English.

## 4.9 Editing PO Files

We anticipate that some gnu.org translators will find this format odd or inconvenient, if they never happened to work with PO files before<sup>2</sup>. Don't worry, you will soon get accustomed to it. It is the established format for translations in the Free World, and if you have any problems, other translators will help you.

The most efficient way to edit a PO file is using a specialized PO editor, because each of them represents and treats gettext messages in a consistent and predictable way. It is possible to edit a PO file with an ordinary plain text editor, but extra effort would be necessary to make the result valid.

Note that recent versions of some PO editors (both offline and web-based) offer access to various translation services that do machine translation for their users. Using a machine translation service is a clear example of SaaS (see *Who does That Server Really Serve?* (<https://www.gnu.org/philosophy/who-does-that-server-really-serve.html>)), so please don't use such editors unless they only submit requests to your (or GNU project's) own servers.

Here is a list of widely used PO editors we can recommend:

- PO mode. We recommend using GNU Emacs in PO mode, because Emacs is the program that is suitable for performing any task when it comes to maintaining the GNU Project's website. Provided that you have GNU gettext installed, any .po file you visit should automatically switch to PO mode. You can enable/disable it with `M-x po-mode RET`. On some GNU/Linux distros such as gNewSense, PO mode is available in a separate package, `gettext-el`. See Section "PO Mode" in *GNU gettext tools*.
- Gtranslator—the GNOME PO editor. See <http://projects.gnome.org/gtranslator/>.
- Lokalize—the KDE 4 editor. See <http://userbase.kde.org/Lokalize>.

---

<sup>2</sup> For detailed information about editing PO files, see Section "Working with PO Files" in *The GNU United Nations Manual*.

- KBabel—the KDE 3 editor. No longer supported, but might be available on some old systems.
- Poedit—another popular editor that is based on the `wxWidgets` graphical toolkit. See <http://www.poedit.net>.
- `po.vim`—ftplugin for the Vim editor. The best option for people who use Vim as their editor. See [http://www.vim.org/scripts/script.php?script\\_id=2530](http://www.vim.org/scripts/script.php?script_id=2530).

## 4.10 Related Mailing Lists

Here is a summary of the mailing lists relevant to the translation process, and a brief description about how they relate to the various participants in the process.

### `www-discuss@gnu.org`

The basic discussion list of the GNU Webmasters. All team leaders are required to subscribe.

This is a private mailing list.

### `www-commits@gnu.org`

Commits to the ‘`www`’ repository are sent here. All Translation Managers are required to subscribe. It is strongly recommended that team leaders subscribe—in any case they should, and mail delivery can be disabled personally.

This is a public mailing list, so everyone can subscribe and review the archives. The ‘`www`’ CVS repository is also public.

### `trans-coord-discuss@gnu.org`

The main discussion list for the GNU Web Translators. Team leaders must subscribe, as errors from GNUN are mailed here. It’s highly recommended that active team members join as well, because the changes in general policies for translations are also announced and discussed here.

This is a private mailing list.

### `trans-coord-news@gnu.org`

This is a list for notifications about GNU United Nations releases. It is not mandatory to subscribe to it, although the traffic is very low. If you want to track only GNUN release announcements, subscribe to the ‘`gnun`’ topic via Mailman’s user interface.

Automatic announcements for new `gnu.org` translations (provided they’re handled by GNUN) are also delivered here. There are separate ‘`lang-ann`’ topics for every GNUN-aware language, so it is a good idea to advertise this capability widely among your local community. For example, if a reader wants to be informed only about new Spanish translations, she can just subscribe to the ‘`es-ann`’ mailing list topic.

This is a public mailing list.

### `trans-coord-devel@gnu.org`

All development of GNUN happens here. Commits to the ‘`trans-coord`’ repository are also sent to this list.

This is a public list, and `bug-gnun@gnu.org` is an alias.

### `webmasters@gnu.org`

This is a tracker for GNU Webmasters. It is used for bug reports and other suggestions for (English) `www.gnu.org` web pages.

This is a private tracker.

### `web-translators@gnu.org`

This is the tracker and the primary contact of GNU Web Translation Managers. It is used for bug reports against `www.gnu.org` translations and submitting new



translations for the languages lacking an active team, requests for help from the teams and various translation-related requests from GNU people.

This is a private tracker.

Every team should also use at least one mailing list on Savannah, see Section 3.5.7 [Savannah Mailing Lists], page 12.

## 4.11 Savannah Project Membership

Participants in the `www.gnu.org` translation process normally have to be members of the following Savannah projects, depending on the case:

**‘www’** The main project which hosts the `gnu.org` Web repository. Administrators are the Chief Webmaster, entrusted webmasters and the Translation Manager (in order to approve leaders’ applications). All team leaders (and co-leaders) should be members of this project.

Note that this project has no direct relation to translators, although almost anything happening in `www` directly affects them. The `www` project is managed separately and has a different (entirely unrelated) process for approving contributors.

**‘trans-coord’**

An organizational project especially created for co-ordination and improvement of the translation process. All team leaders are required to be members, as bugs reported to `web-translators@gnu.org` are often redirected to the `trans-coord` ‘Bugs’ tracker.

The admins of this project are the GNU Web Translation Managers.

**‘www-lang’**

All translation team leaders of the language *lang* should be admins of the project `www-lang`. The leaders may also appoint some other members as `www-lang` admins for team’s internal reasons.

## 4.12 Summary of SSI #includes

The GNU Project’s website uses SSI (Server Side Includes) to manage some common parts that are the same in many of the articles. With the help of GNUN their handling should be behind the scenes, but for some of them manual intervention is needed. Here is an incomplete list of the `#include`’s used:

`server/banner.html`

This file contains only `#include` directives, so the “translation” should be almost identical, with filenames modified to have the *lang* extension. The only other difference is including `server/top-addendum.lang.html` at the end.

`server/body-include-1.html`

Contains the top menu with useful “skip to” links.

`server/body-include-2.html`

This is the file containing the menus, the FSF widget, and any visible announcements made from time to time. If a string gets “fuzzy” or “new” here, it will appear in English in all translations, until `server/po/body-include-2.lang.po` is updated. Note that some validation errors originate from an error in `server/body-include-2.lang.html` or some other template file.

`server/bottom-notes.html`

A link to the FSF page explaining how to report possible copyright infringements.

**server/footer-text.html**

This is a short file currently containing the footer links, the FSF mission statement and the “back to top” link.

**server/generic.html**

This file is empty; its “localized” versions may contain optional short messages providing more information about the translation team or where to report bugs.

```
<p>To join the Fooish translation team, see <a
href="https://www.gnu.org/server/standards/translations/foo">the
Foo team homepage</a>.</p>
```

This file is not under GNUN’s control, you should edit HTML directly.

**server/header.html**

The declaration that is included in literally every file. It is maintained manually, as it does not make much sense to put it under GNUN’s control (there are no translatable strings). Remember to specify the proper `xml:lang` and `lang` attributes, and for RTL languages, the `dir` attribute. For example, the file `header.ar.html` should contain this line:

```
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="ar" lang="ar"
dir="rtl">
```

**server/head-include-1.html**

This file (included from `server/header.html`) is very important: the encoding is defined here. Even if a specific PO file is deliberately encoded in another encoding, the generated HTML will contain the encoding declared in the `<meta>` element at `server/head-include-1.lang.html`, so browsers will obey it.

The encoding must be UTF-8, because the English text in the “no-grace” articles serves as a replacement of the translation when the latter is not complete, and because all translated pages share automatically generated lists of translations.

**server/html5-header.html**

This file was included in pages using some entities introduced in HTML5 draft. These days this is the case for all pages, so the file redirects to `server/header.html`.

**server/html5-head-include-1.html**

Likewise.

**server/head-include-2.html**

Imports the standard CSS, which can be overridden. See Section 4.14 [CSS], page 23.

**server/home-pkgblurbs.html**

This header includes short descriptions of all GNU packages; it is included from the homepage and `manual/blurbs.html`.

**server/footer.html**

This is a very short and simple file, containing another `#include` directive. It is maintained manually, so just add `lang` to the filename, in order the localized `footer-text.lang.html` to be included.

**server/outdated.html**

This file is automatically included in outdated translations. It contains a message with links to the English file and to a generated difference of the current revision of the English file against the most recent revision that has a complete translation. It is only included in articles affected by “grace period” because in those cases the outdated passages are replaced with English text, and it is evident without any notices that there is no complete and up to date translation.

`planetfeeds.html`

Includes automatically extracted news items.

`server/top-addendum.html`

The text saying that the page is a translation.

`licenses/gpl-3.0-body.html`

`licenses/fdl-1.3-body.html`

... Some of the licenses have the text of the license itself separated in another file. This serves two purposes: 1) to provide a “standalone” HTML version of the license without the gnu.org style; 2) to prevent strings sneaking in the `.pot` files, as licenses have only unofficial translations, hosted elsewhere. Nothing special should be done about these SSI directives; the files generated by GNUN include them verbatim as they should not be translated.

The files

- `header.html`
- `head-include-1.html`
- `html5-header.html`
- `html5-head-include-1.html`
- `head-include-2.html`
- `banner.html`
- `body-include-1.html`
- `body-include-2.html`
- `bottom-notes.html`
- `footer.html`
- `footer-text.html`

in the `server` sub-directory are what webmasters call “the server templates”. These files are included in almost every article, translated or not. They are somewhat important, as an error made in translating them may break every translated page. The server templates and the homepages are rebuilt by GNUN whenever the original English files change; the `GRACE` variable has no effect on them. See Section “Runtime Variables” in *The GNUUnited Nations Manual*.

### 4.13 Technical Pages

These pages make the localization of `www.gnu.org` complete, so you may want to translate them even though they are not on the philosophical priority list.

The first one is `gnu-404.html`. This is the page shown when the visitor encounters a broken link; it explains the situation and suggests a few frequently requested pages.

The second page is `server/select-language.html`. It explains how the language negotiation works (see [language-negotiation], page 18) and provides a way to customize it to some extent.

### 4.14 How to Use Custom CSS

The CSS file `layout.css` gets included (with three other CSS files) in almost all the English articles through `server/head-include-2.lang.html`. However, sometimes this style isn’t quite right for translations—many languages have much longer expressions, and that is natural. To include your own CSS, create a file `style.lang.css` and add it *after* the directive to include `server/head-include-2.lang.html` and *before* the closing `</head>` tag in `server/banner.lang.html`, i.e.

```
<!-- start of banner.bg.html -->
```

```
<!--#include virtual="/server/head-include-2.bg.html" -->
<link rel="stylesheet" href="/style.bg.css" media="screen" />
</head>
```

Override only what is necessary and looks broken in your language; do not invent your own style. This is important for the consistency of the gnu.org website. Also, please check if the issue is language-independent; in this case a change for `layout.css` should be discussed with the webmasters.

A typical language-specific `style.lang.css` file looks like this:

```
.inner { max-width: 85em; }

#fssbox {font-size: 50%;}
```

This widens the menu and the area where the articles are displayed (because the menu entries are *much* longer than the English equivalents when translated), includes a localized logo, and makes the font size for the FSF widget twice smaller (because in this language, the translations are almost twice longer and displayed truncated, which is undesirable).

When creating your own `style.lang.css`, don't forget to include the license notice from the `layout.css`, with a short comment.

If using the default CSS style for translations does not give the expected good results, or there are other problems (significant or not) that obstruct reading or worsen the look from an aesthetic point of view, please write to [webmasters@gnu.org](mailto:webmasters@gnu.org) with a description of the issue. If there are several unrelated problems, send separate messages with appropriate explanation (which may include a demonstration of the bug, such as a screenshot).

#### 4.14.1 Specific Issues Related to RTL

Unfortunately, the <https://www.gnu.org> website does not have excellent support for languages using right-to-left scripts, although best efforts are made. If your language is in this category, make sure to:

- Set the attribute `dir="rtl"` in the `html` element at `server/header.lang.html`.
- You must include an additional CSS, `style.rtl.css`, to override some of the pre-defined values. See template files for Arabic and Farsi to understand how these two languages solve some of the problems. See Section 4.14 [CSS], page 23.

**Important:** Some articles contain their own `<style>` redefinitions, or style attributes in the form `<p style="...">`. In such situations, it is quite possible that the general language-specific CSS does not help, and the translation of this specific article does not look correct. Please write to [webmasters@gnu.org](mailto:webmasters@gnu.org); if you have a working solution that works for both cases—so much the better. For general issues that affect your language and require a general solution, write to [webmasters@gnu.org](mailto:webmasters@gnu.org) as well, precisely describing the problem.

### 4.15 Migration to the New Style

Migration to the new style should be straightforward, and this is one of the problems GNUN set out to solve. If you have to migrate old-style translations, see Section “Migrating” in *The GNU United Nations Manual*. If the old translation is HTML 2.0 (or 3.2), you still have to take care about the inner markup. Overall, it is substantially easier than doing all of it manually.

# Appendix A GNU Free Documentation License

Version 1.3, 3 November 2008

Copyright © 2000, 2001, 2002, 2007, 2008 Free Software Foundation, Inc.  
<http://fsf.org/>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

## 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. Secondly, 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

- - .odt files ..... 3
- C**
- capitalization ..... 17
  - checking out web pages ..... 3
  - co-leaders ..... 12
  - co-ordinators ..... 5
  - contacting team ..... 2
  - copyright notices ..... 18
  - Creative Commons ..... 18
  - custom CSS ..... 23
  - CVS, using ..... 9
- D**
- distribution terms ..... 18
- E**
- Emacs ..... 19
  - encoding ..... 3, 22
  - external resources ..... 10
- G**
- generic.html ..... 22
  - gnu-404.html ..... 23
  - GNUN ..... 3
  - Gtranslator ..... 19
- H**
- HTML5 ..... 22
- I**
- initial server templates ..... 6
  - internal links ..... 18
- L**
- language code ..... 6
  - language negotiation ..... 18, 23
  - links, internal ..... 18
- M**
- mailing lists ..... 20
- N**
- native English speakers ..... 7
  - news items ..... 22
- O**
- ODF ..... 3
  - outdated translations ..... 22
  - outdated translations, notification ..... 16
- P**
- plain text ..... 3
  - planetfeeds.html ..... 22
  - PO editors ..... 19
  - PO, editing ..... 19
- R**
- review ..... 7
  - right-to-left languages ..... 24
- S**
- SaaS ..... 19
  - Savannah account ..... 5
  - Savannah projects ..... 21
  - Savannah, registering ..... 5
  - server templates ..... 3, 23
  - server templates, initial ..... 6
  - server/language.html ..... 23
  - SourceForge ..... 9
  - SSI includes ..... 21
- T**
- tasks, tracking ..... 8
  - team homepage ..... 11
  - team leaders ..... 5
  - team mailing lists ..... 12
  - team members ..... 2
  - team news ..... 12
  - team status ..... 11, 13
  - team's repository ..... 12
  - team, contacting ..... 2
  - technical pages ..... 23
  - tracking changes in files ..... 16, 20
  - tracking tasks ..... 8
  - trans-coord-discuss@gnu.org ..... 20
  - translation priorities ..... 14, 16
  - Translations README ..... 2, 17
  - translations, keeping current ..... 16
  - translations, review ..... 7
  - translations, unreviewed ..... 9
- U**
- unreviewed translations ..... 9
- V**
- VCS ..... 12

**W**

web pages, checking out .....	3	web-translators@gnu.org .....	20
		webmasters@gnu.org .....	20
		www-discuss@gnu.org .....	20