

How to Contribute Without Being a Programmer?

Become a FreeBSD Translator!

BY DANILO G. BAIO



The FreeBSD Project is adopting Weblate as its web-based, continuous, localization platform, which makes it easy for people to become translators. It began operation at the start of this year and so far has gathered 10 languages and 89 contributors to help translate the FreeBSD documents.

Language	Code
Chinese (Simplified)	zh_CN
Chinese (Traditional)	zh_TW
French	fr_FR
German	de_DE
Italian	it_IT
Norwegian	nb_NO
Persian	fa_IR
Portuguese	pt_BR
Spanish	es_ES
Turkish	tr_TR

The list shows only the current languages in operation on our Weblate instance—new languages are very welcome. This list doesn't show the current translations that are already committed in the FreeBSD Doc repository.

With Weblate, it's so easy to translate that contributors don't need to handle a repository, an xml file, builds, or to understand DocBook/XHTML. They just need to learn some cool things and translate it to their native language in the process.

Focused on learning, the FreeBSD Documentation is very known for its quality. When you begin to translate any document, I can assure you that you will understand why.

The full set of documents contains 35 articles and 8 books. Here are the current statistics from Weblate (<https://translate-dev.freebsd.org/projects/freebsd-doc/#information>).

	Strings	Words
All documents	39,779	691,513

The *FreeBSD Handbook* is a large document and currently the main document of the Project.

	Strings/sentences	Words
FreeBSD Handbook	12,212	254,554

There are smaller documents as well—articles for example:

	Strings/sentences	Words
Contributing to FreeBSD	217	5,622

They all can be accessed on <https://www.freebsd.org/docs/books.html>.

For new languages, we recommend starting with one or two small documents before heading to the *FreeBSD Handbook* so that people can get used with the tool and also to see their work committed faster.

When translating you need to give your full attention to the text, understand the context, and occasionally re-read it, but that's a good thing and it's how you will learn in the process. I still remember some specific details of things I translated in the past.

Simple steps to start contributing:

- **Create an account, with an email address or with your Github id**

<https://translate-dev.freebsd.org/>

Weblate creates new commits for each translation and references them with your email address, so it's easy for you to track your work. Your translations will appear on your Github Contribution Activity so long as you use an email address that matches with your Github account.

- **Subscribe to the freebsd-translators mailing list**

<https://lists.freebsd.org/mailman/listinfo/freebsd-translators>

Updates in the infrastructure and all news about translations are sent there. If you have any doubt, that's the place for you to check.

- **Introduce yourself in the mailing list and ask to join a language team**

If the language team does not exist, ask to create it—you could be the coordinator.

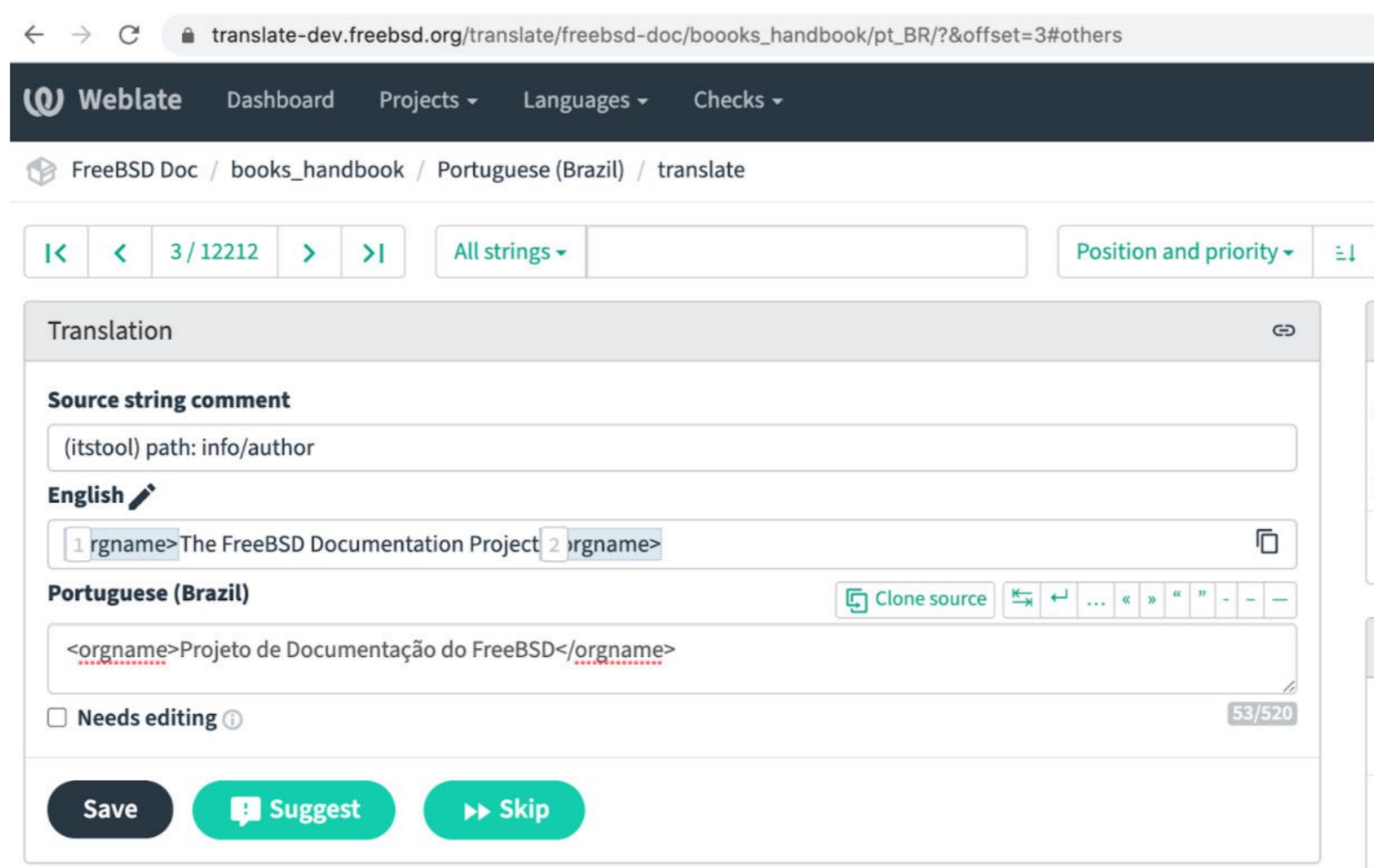
The introduction is important because you can discover other people translating into the same language and make it easier to coordinate efforts.

- **Log in to Weblate with your new account**

After your introduction, an admin or coordinator will add your username to a language team and you can start translating.

- **Find your language team and choose an initial document to translate**

Translating online on the Weblate interface is the easiest way to contribute, once a coordinator or admin has given you access to a language. The save button will be enabled and you can start working.

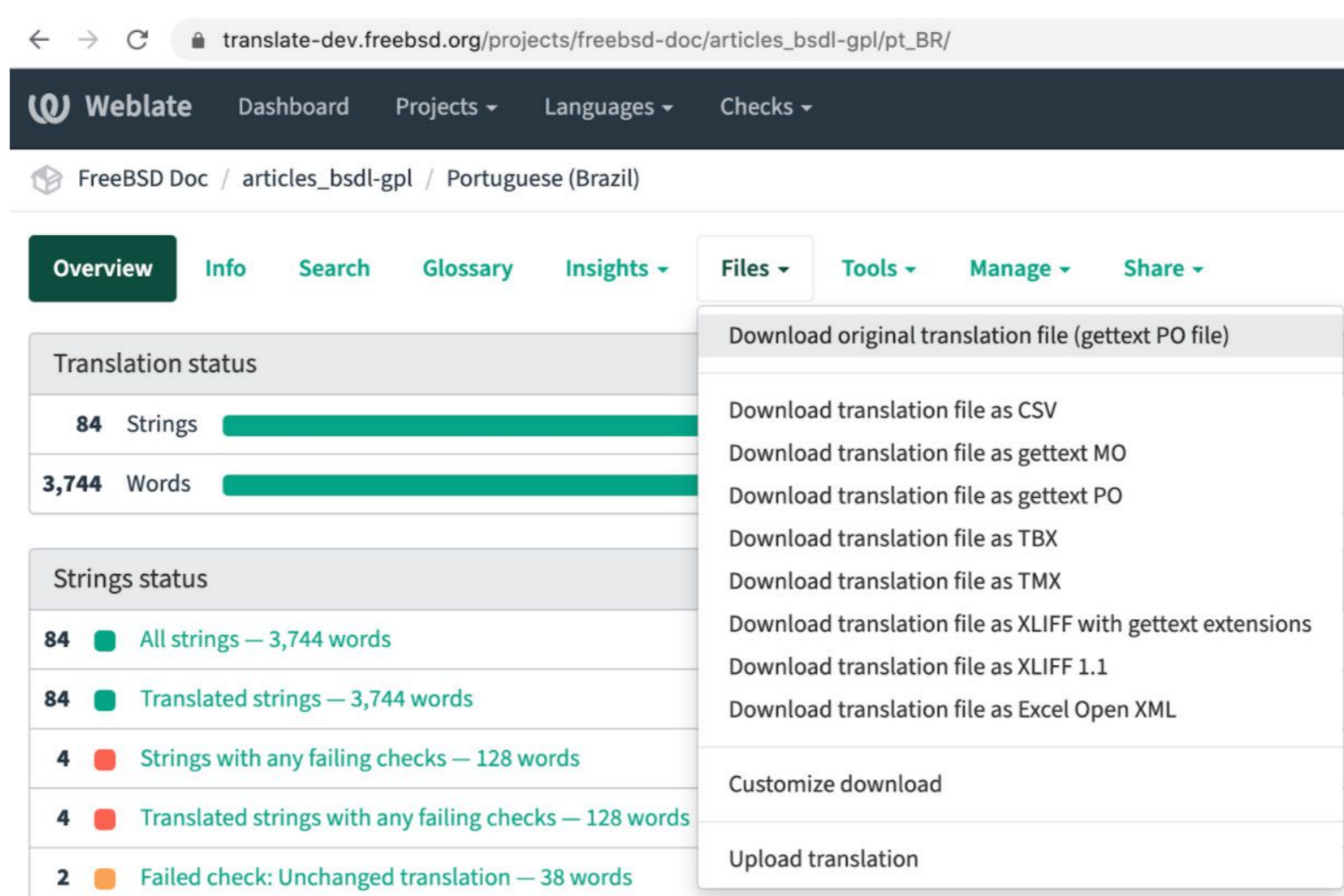


Weblate has a set of links that lead to actual translation. The translation is further divided into individual checks, like *Untranslated* or *Needing review*. If the whole document is translated, without error, the *All translations* link is still available. Alternatively, you can use the search field to find a specific string or term. You can find out more info about translations in the official Weblate documentation—things like keyboard shortcuts and other tips about the translation tool, but the interface is very intuitive.

<https://docs.weblate.org/en/latest/user/translating.html#translation-projects>

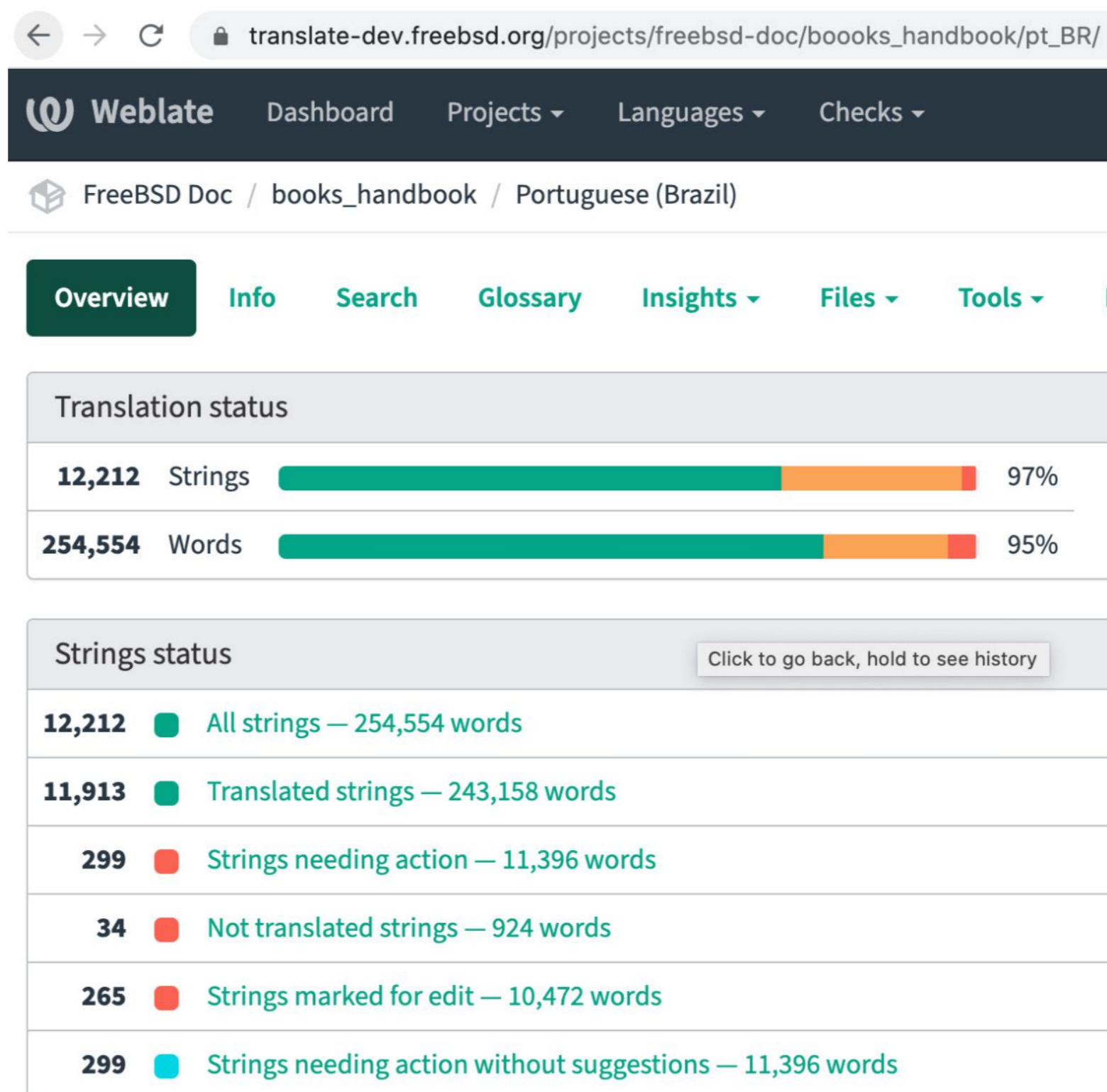
Translating offline

If you are familiar with PO Gettext and would like to make offline translations, you can download and upload your translations through the document page of your language by clicking in the Files section.



Proofreading and Weblate Quality Checks

When you click on *Project / Language / Document*, the Translation and String Status from Weblate for that document will be shown. This page is very useful for proofreading and quality checks.

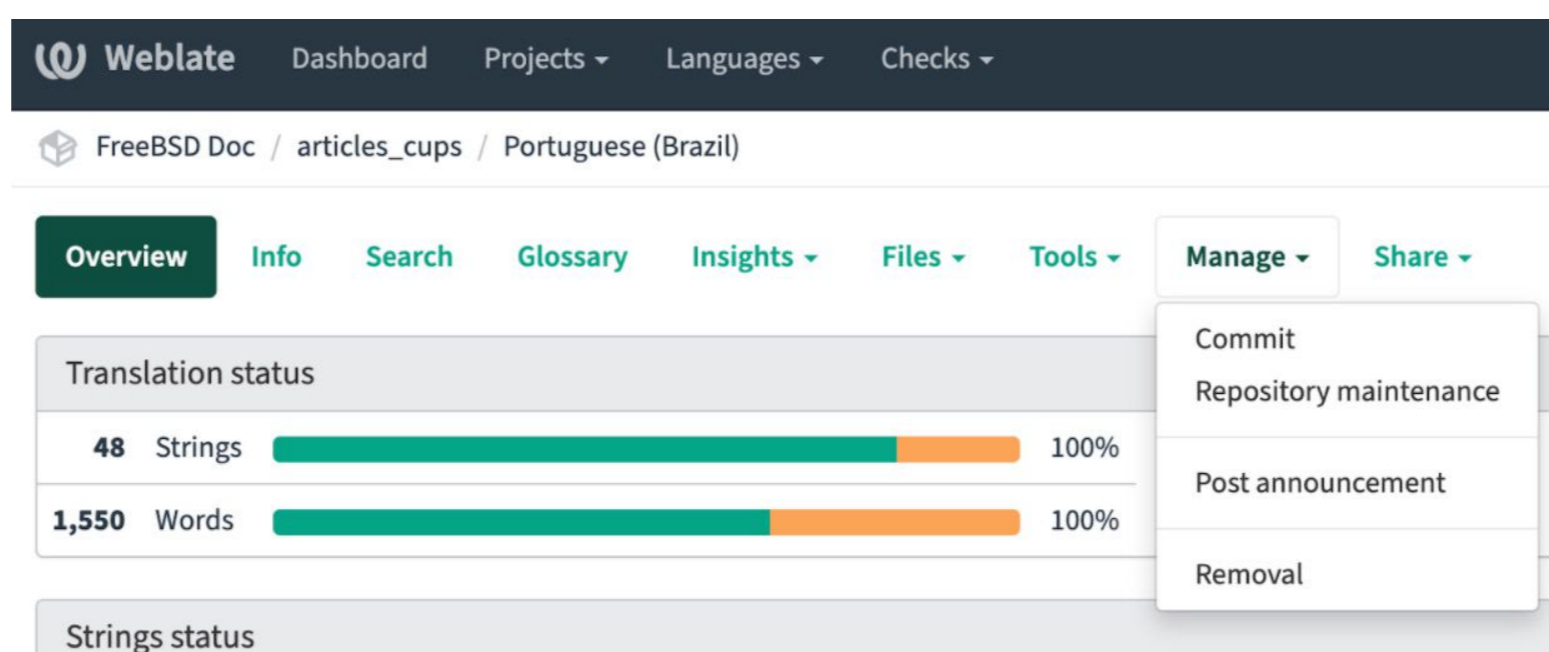


In this example, there are some strings needing revision. If you click on these links, it will show only those strings that need to be revised/translated.

It is often more helpful to see the translated strings in their final context in order to do the proofreading more easily, and our project has some facilities for this.

Building the Translated Document

All translations in our Weblate instance are built daily and they are available on <https://doc.fugbr.org/jenkins/>. We enable these builds for all new languages. If you want to speed up this process, you can tell Weblate to commit your translations any time you want and then a build process will be triggered immediately through our Jenkins, and some minutes later you can proofread it.



You can also build the documents locally. There is more information about that on our wiki page on <https://wiki.freebsd.org/DocTranslationOnWeblate>.

Submitting Translations

After finishing any document translation, submittal will be the last step to push your work officially to the FreeBSD website.

In the book *FreeBSD Documentation Project Primer for New Contributors*, there is a chapter explaining how to submit translations. It is not difficult and if this is your first document, we will have people help you in the process. https://www.freebsd.org/doc/en_US.ISO8859-1/books/fdp-primer/po-translations-submitting.html

After committing translations to the official documentation tree, all translators' names will be added to the list of *Additional FreeBSD Contributors*. Congratulations! Now you are part of the FreeBSD history as well.

Final Thoughts

For translating, you don't need to be a developer, you just need to be willing to contribute, and as noted before, you will learn some new things in the process.

There are developers who started contributing to FreeBSD Documentation and later become ports or kernel developers, this can be an initial journey for you in the Project.

In the FreeBSD Documentation, subjects are broader and there are pieces of Unix and BSD history there, stories about amazing contributors who sadly aren't with us anymore. For instance, read the Bruce D. Evans In Memoriam text—it's in the article *Contributors to FreeBSD*, chapter *Development Team: In Memoriam*, which I only became aware of when translating it.

It's really nice to know the history and be part of it as well. To me, being part of the FreeBSD Project is such an honor.

If you have any questions, join us on `freebsd-translators@` mailing list. You will be very welcome. Please help promote this effort to your local user group as we always need more volunteers.

DANILO G. BAILO is a FreeBSD ports committer and Brazilian Portuguese translator.
dbaio@FreeBSD.org

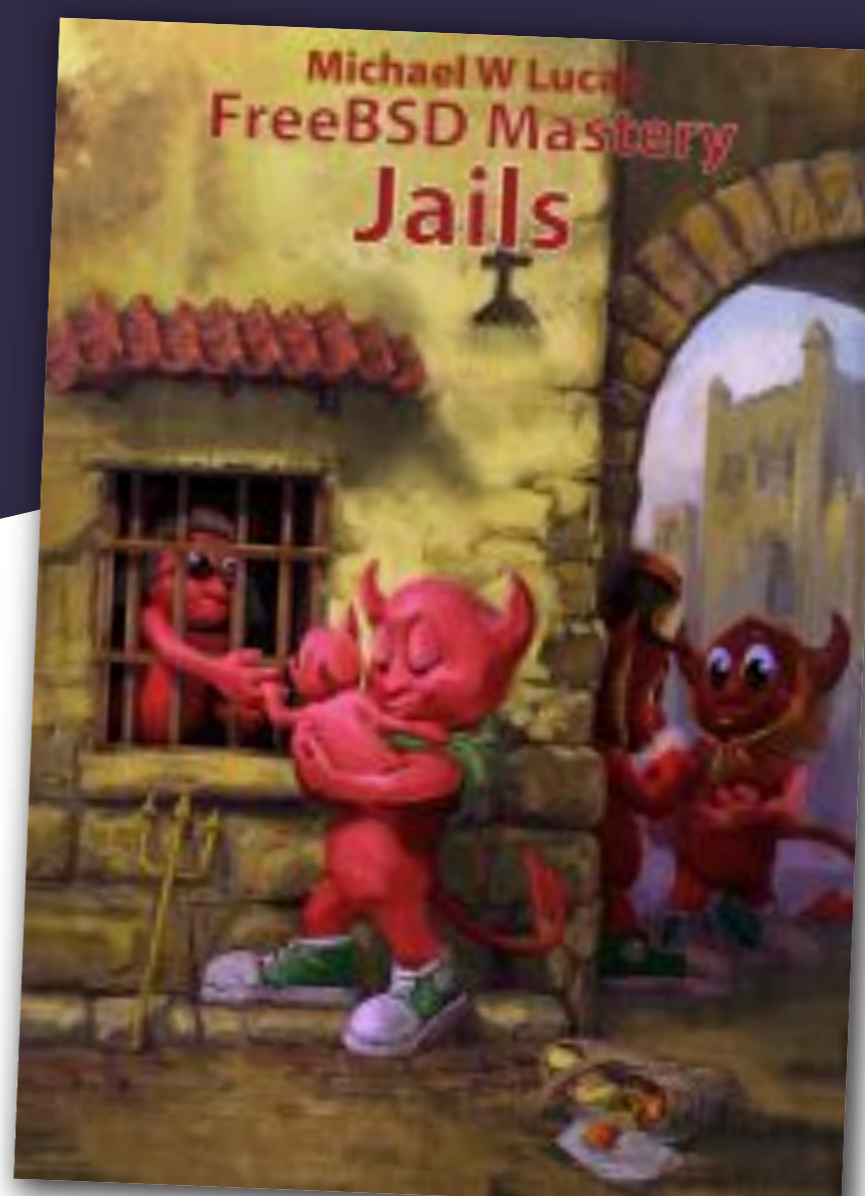
Jails ARE FreeBSD'S MOST LEGENDARY FEATURE:

KNOWN TO BE POWERFUL, TRICKY TO MASTER,
AND CLOAKED IN DECADES OF DUBIOUS LORE.

FreeBSD Mastery: Jails cuts through the clutter to expose the inner mechanisms of jails and unleash their power in your service.

Confine Your Software!

- * Understand how jails achieve lightweight virtualization
- * Understand the base system's jail tools and the iocage toolkit
- * Optimally configure hardware
- * Manage jails from the host and from within the jail
- * Optimize disk space usage to support thousands of jails
- * Comfortably work within the limits of jails
- * Implement fine-grained control of jail features
- * Build virtual networks
- * Deploy hierarchical jails
- * Constrain jail resource usage.
- ...**And much, much more!**



FreeBSD Mastery: Jails BY MICHAEL W LUCAS **Available at All Bookstores**