

Localization and Drupal

What you need to know about software
localization, including Drupal

by Rūdolfs Mazurs

About me

- New to Drupal community
- Translating software is my hobby
- Have translated OpenOffice (now LibreOffice), GNOME and Ubuntu projects.

Why do it?

- No one else can do it!
- It makes the software accessible to more people
- Users will fear less to use it (at least will make less errors)
- Website will look more professional
- It is fun!

What needs to be known

- How does the process work
- Computer assisted translation CAT
- Glossary
- Translation memory (TM)
- Variables and plurals
- Spelling, grammar and style
- Quality assurance

Process overview

- Get the source files (.po) from <http://localize.drupal.org/>
- Translate the strings in the files
- Send the result to the maintainer

OR

- Gather local community
- Have fun and translate
- Let the community leader handle the technicalities

Translation software

- Offline
 - Lokalize (most powerful)
 - poEdit
 - Virtaal
- Online
 - Pootle
 - Transifex
 - Launchpad

Glossary

- Same things are called same way
- Different things are called differently
- Use the already established terms first
- Write down most important terms (message, post, username, etc.)

Translation memory

- Remembers all the previous translations
- Suggests new translations
- Helps to keep terminology in check
- Makes mass renaming easier

Variables and plurals

- Find out, how variables are written. In Drupal, numbers start with “@” symbol, strings with “%”
- Plural forms help to translate strings, depending on object count. E.g. “1 apple”, “4 apples”. There are different plural forms for different languages. Explanations can be found here http://unicode.org/repos/cldr-tmp/trunk/diff/supplemental/language_plural_rules.html
- In case you need the plural form codes, they can be found here <http://translate.sourceforge.net/wiki/l10n/pluralforms>

Spelling, style and grammar

- Spell check — use it, but don't trust it
- Rules of grammar don't disappear
- Style must be appropriate and consistent (e.g. whole project should be either formal or informal)
- Avoid to translate word by word. Even if sentence is technically correct, it may sound weird.

QA

- Translations should be checked by others
- Most important terms should be checked against translation memory
- List of typical mistakes should be kept and checked against
- Testing in user interface (web site), since there is no context in the .po files
- Report all problems (spelling, missing plural forms) in original texts to developers