

Translation in ERP5

There are translation domains in ERP5:

1. erp5_ui - Use Localizer's predefined translation for hard-coded messages
2. erp5_content - Use Localizer's predefined translation for user's input
3. content translation - Use user-defined translation for a specific document's specific property value entered by user

Localizer is useful to translate a phrase widely used in the system, in other words, it will give the same unified translation across the whole ERP5 for a specific phrase. Localizer's predefined translation system is the best.

Content translation is useful to translate user input which can't be predefined by an ERP5 developer. If a user wants to input a phrase by her/himself and if the phrase has to be translated (for example, product name, person name etc), then content translation is a solution allowing the user to provide translations by her/himself.

Translation Tools and Standard Procedure

This chapter defines the standard translation procedure. It does not explain how to use the tools to translate, but introduces them and explains their purpose. The following chapters in this document will explain how to use those different tools.

The Standard Translation Procedure

This section defines the standard translation procedure. It lists the different steps of the Translation Procedure, from the Translation itself to testing the Translation.

Translation

This paragraph explains the different steps to take in order to translate ERP5, based on different tools that will be explained in the next paragraph. There are 5 steps:

- Check and improve the Glossary entries;
- Translate the Glossary;
- Translate .po a file generated by ERP5 instance;
- Finalize work with Localizer;
- Commit the .po file into a business template.

Let us review the 5 steps one by one.

Check and improve the Glossary entries([Full list of Glossary entries](#)). This step is essential because it will help reaching a consistent English version of ERP5. The glossary contains a limited set of terms which are specific to ERP5 and to a given business field. This step is necessary to make sure that every term used in ERP5 is well chosen, well defined and well understood. This step is so important that it should actually be included as a compulsory step in a development process of new ERP5 modules. This step requires to involve ERP5 developers to clarify terms whenever needed.

Translate the Glossary. Based on the clear description of every term in the English glossary, it is possible to translate all glossary terms and their description to another language. The translated glossary will be used as reference information by translators in charge of translating the hundreds messages of ERP5. Spending time on the glossary translation is required to improve translation consistency.

Translate the .po file. A large .po file is generated from ERP5. It contains all user interface messages to translate. Part of this .po file can be translated automatically using the glossary and a helper tool. The rest of this .po file must be translated by a human translator. The use of a .po file at this step is useful to benefit from numerous GUI tools and translation memories. This step leads to a rough global translation, based on a consistent Glossary.

Commit the translation business template. A business template is created with a rough translation and committed so that it can be shared, finalized and tested.

Finalize work with the Localizer. ERP5 translation mode allows to review the translation in its real context: forms, modules, reports, etc. This step is useful to detect errors in the rough translation, fix details and improve translation quality.

Testing the translation

Once all the steps of the standard Translation Procedure have been taken, the Translation must be tested before it is

released to final customers. To do so, it will have to pass the following set of tests, as described in this document [\[AD\]](#).

- Make sure there are no different Titles for identical entries in the Glossary;
- Make sure each term of a given business field has proper Title and Description in the English Glossary;
- A skilled person in the local language and in the business field makes sure that the terms are well translated in the local language Glossary;
- An ERP5 script makes sure that all terms are translated (Completeness Test)
- Application developers test the translation before releasing it (Internal Test)
- A panel of external users tests the translation (External Test)

Once all tests have been taken successfully, the translation is considered as "Good" as described in the Best Practice document [\[AD\]](#), and can be released to final customers.

Translation Tools

This section describes the various translation tools which are needed to Translate ERP5: the Glossary, the .po file generator and the Localizer.

The Glossary

The Glossary gathers all specific terms which are used to build user interface messages in ERP5. The number of glossary terms is quite limited. The goal of the glossary is not to gather all messages but rather to gather a few terms which need to be perfectly chosen and documented in order to produce a usable user interface in ERP5.

Each term from the Glossary must be well chosen. This means that the chosen word must leave a little room as possible for ambiguity.

Each term from the Glossary must be well documented. This means that it must have a detailed description which does not leave any room for fuzziness or doubt.

The .po file generators

The .po file is generated by a script in ERP5. This script extracts all user interface messages of ERP5. Another .po file can also be generated from the Glossary and merged with this file in order to accelerate the translation process by filling some messages with terms already present in the Glossary. The use of .po file is useful to quickly produce a complete translation. However, this translation may lack some context information and will require extra work.

The Localizer

The Localizer is the tool which helps translating live ERP5 applications. With Localizer, it is possible to click on terms displayed in forms and translate them interactively. Using the Localizer is useful to finalize a translation and to fix mistakes which were not obvious in the .po file and which become obvious in the context of the live application.

Working on the Glossary

This chapter explains the basics of the Glossary module. It explains how to improve it, how to use it for translation and, finally, how to extract .po files from an ERP5 site and from the Glossary.

Glossary Basics

The forge business template [\[RD\]](#) provides a Glossary module to help ERP5 translators. The purpose of the glossary module is to gather a few core terms present in the user interface messages and to define each term precisely in the context of a business field. The Glossary module helps translator to produce a "good" translation as defined in the Best Practice document [\[AD\]](#).

The glossary module is an ERP5 module which contains a collection of so-called "Term" documents. Each Term document defines the relation between a technical term in ERP5, which is common to multiple business fields, and a well chosen term which is specific to a business field.

The structure of a Term document is described below:

- **reference**: the technical term in ERP5, such as a property or a workflow state.
- **business field**: the name of the context in which the term is used. This can be for example "accounting", "crm" or a child context of a general field such as "crm/ticket".
- **language**: the ISO abbreviation of language name such as "en" for English.
- **title**: the non ambiguous chosen term which will be used in all messages displayed to the user. If the language is "en", then the title is the reference term which must be used by all forms in ERP5, since English is the reference language. If

the the language is "ja", then the title is the Japanese translation of the original English term.

- **description:** a is a precise description description which explains the meaning of the chosen term in the context of the business field. This description helps users, translators and developers understand the meaning of the term. The description will be displayed in the Glossary Module list of terms, and in the .po file that will can be extracted from the Glossary. This Description field is really important because it helps translators understand the context information of a given term. Descriptions will also sometimes be displayed in the application as a pop-up help messages.

Glossary Improvements

This paragraph explains how the original English Glossary can be improved. Improving English terms is important in order to reduce ambiguities in chosen terms and improve their description to help translators.

Reviewing and editing original English terms

Editing a term consists of changing its Title and Description. Once you are satisfied with a term, you can validate it through a workflow action. Open the "Action" menu item list and choose "Validate Workflow" action. This way, you confirm to other translators that this term is appropriate.

In order to help you review the original English Terms of the Glossary, you can use different tools like dictionaries or online applications. A list of helper web sites is provided on the ERP5 web site [\[RD\]](#).

Modifications of English Terms in the Glossary need to be reflected in ERP5 forms, workflow, etc. to be taken into account. This process can be greatly accelerated thanks to helper tools provided at the level of the Glossary module. Just follow this procedure: select "Update Fields by English Glossary" in the Action menu item list of the Glossary module. A dialog will appear, asking you to select one or several business fields you want to update terms of. Click on the "Compare" button to display the list of fields which are going to be updated. Use this list to select appropriate titles and descriptions term by term.

The "Note" column sometimes contains the value "Delegated". This value means that the Title of the field is acquired from another field through delegation and that the field does not hold its own title value. In such case, it is better not to change the field and rather update the proxy target field instead of the field itself. If the Note column contains the message "Tales is used", then update is usually without effect on the user besides consistency for developers.

Help documents for translation

Translation is a complex procedure which requires attention and precision. In order to help you in this procedure, it is strongly recommended to read those documents before you should start the Translation:

- Translation Best Practice [\[AD\]](#). This documents defines a "good" translation and provides a list of Best Practices for translators along with providing common translation crimes and pitfalls to avoid.

Translation with glossary

This paragraph explains how to translate the terms from the Glossary module. Be aware that this step must be taken only once the original English Glossary has been proven to be complete, precise and documented. Translating a Glossary filled with badly chosen or inappropriate original terms will only lead to losing time.

Preparation

The Glossary is divided into business fields. With this approach, it is possible to assign each translator to one or more business fields and subdivide work. Please note that a translator has to be skilled or at least has to have a basic knowledge of the business field he or she is assigned to translate, as business fields gather technical terms like accounting terms or CRM terms. Many dictionaries and glossaries are available on the Internet to help translators. If you find more online resources useful to translators, please add them to the resource section on the Documentation Localisation section [\[RD\]](#).

Translating the English Original Terms from the Glossary

Once the English Glossary is perfect, and only at this time, you can start the translation of the English Original Terms of the Glossary into a foreign language. The Glossary module has a feature to generate untranslated terms in a foreign language from existing English ones. It is useful when you start translation. In order to do this, follow this procedure:

Select *Create Terms from English Ones for Translation* in the Action item list. Select the language you want to create the terms of, and click "Create Terms" button. This action will create an empty term in your local requested language for each English Original Term of the Glossary.

Once you have created the terms in your local language, you can start the Translation of the Glossary following this procedure:

- Choose an untranslated term in your local language and click on it.

- In an other window/tab open the equivalent original English Term in order to read the description of the term.
- Translate the Title and Description fields of the term in your local language.
- Repeat these steps until all the terms from the Glossary have been translated in your local language.

Refer to the referenced and applicable documents in order to learn to best translation practices and translation crimes. Reading those documents will help you save much time and reach a better translation quality faster. Keep in my that your translations will be later tested thanks to a test procedure.

Generate two .po files

This paragraph explains the last step of the Glossary Translation. After all the terms of the Glossary have been translated in your local language, you have to export .po files from this Glossary. Two different .po files have to be extracted: the first one gathers all the terms that you have translated in your local language from English Original Terms, and the second .po file contains Notification Messages from the different installed business templates, as described in below

Extraction of local terms .po file

To export the translated local terms to a .po file, click the import/export icon of the Glossary Module. This will open a new dialogue in which you can select the different business fields and languages you want to export. Make your choice and click the "Export Terms As PO File" button.

Extraction of static Notification Messages of Business Templates

The local terms are not the only resource to be translated in order to achieve a complete translation. In business templates, there are many other sources that have to be translated:

- Skins (Python Script, Page Template, ERP5 Form),
- Actions from various action provider tools (portal_actions, etc),
- Workflows,
- Portal type.

You can use the "translation message extraction script" that erp5_forge provides. The script will find static translation messages from installed business templates, and extract them to a .po file, which you will have to merge with the local terms .po file and create a unique .po file to be processed.

In order to extract those messages to a .po file, select "Generate POT file" in the Favourite item list. The messages extracted this way must be translated as well. Once this .po file is extracted, you have two different .po files that you will have to merge into a unique file following the procedure described in the next chapter of this document.

Glossary History

The Glossary origin and structure itself proved to be difficult when translating ERP5.

Glossary Origin

According to the technical team of ERP5, the Glossary has been built backwards. Indeed, it has been extracted from the terms of the application, and it seems that the common sense would have recommended the opposite, which is built the Glossary first and built ERP5 based on this built Glossary.

As different people have been developing ERP5, without Glossary, there are some inconsistency in the terms that are used in English in ERP5. This results in duplicated terms, and a Glossary that must be cleaned up, which is quite a loss of time.

Glossary Structure

While thinking about how the translation should be tested, it appeared that the structure of the Glossary was not adequate:

Portal type is lacking: the absence of a column *Portal Type* in the Glossary results in bad context and then bad translations, and duplicated terms. The portal_type will be useful in the first test that will have to be taken, as described in the [How To Test Translations](#) document.

Lack of clear translation process

This section explains why a lacking translation procedure has created a mess in the translation. It will also list the different documents that have been created in order to provide translators with a good translation process to follow, in order not to make the same mistakes again.

Lack of experience

The lack of experience has been a problem, but as this was the first translation attempt, there could not have been any. Experience is shared in this document. But this lack of experience has led us to lose considerable time, especially the lack of experience regarding proper translation tools, the Localizer and the PO files:

- The **Localizer** is useful because it helps translate live, which means see the result of the translation in the application. It is then really appropriate when the context is lacking. The good process is to browse the application, and when a term is not translated, reach the localizer, search for the term and translate it.
- The **PO files** are useful to see the completeness of a translation, as it will help you see the fuzzy and untranslated terms, which is not the case of the Localizer. On the other hand, it will provide you with less context than the Localizer.

Lack of testing procedure

For a long time there was no testing procedure for translation, e.g. no means to check the quality of the translation. The French translation has been released, even if not complete and not accurate. Obviously it has not been accepted by users either, because many of them complained that ERP5 was not translated.

Fuzzy Assignments

This chapter explains why assignments and organisation issues are part of the translation mistakes that have been done by the Nexedi team. This paragraph will describe these mistakes in order not to do them again.

Who does what?

We have been many to work on the French translation, but in the first attempts, without knowing exactly who was doing what, e.g. who was translating this or that module, etc. This means that instead of cooperating, by dividing the PO files into the proper number of translators, a unique PO file has been generated and worked on by different people of the team. But here is the problem:

- **What to translate?** If someone is told to translate a PO file for example, he or she will not only translate the fuzzy and untranslated terms, but will also check the other terms, and sometimes will change a term that he or she think is not appropriate or bad translated. A good assignment on a PO file is then to split a PO file and ask a team member to translate the entire PO file.
- **Inconsistency:** as the translators were working at random on the same PO file, without being assigned a specific module to translate, there have been some inconsistency in the translations. Indeed, we found out that some PO files had been processed by different persons from the team, resulting in inconsistency while merging the files. This mistake is not so current, and has been done maybe one time, and only during the first translation attempt, but needed to be mentioned here.

Developers/non developers

ERP5 is a complex application containing many complicated/specific terms that only skilled persons can translate. As the Glossary was sometimes lacking descriptions or context, or the context was explained in a technical way, non-developers translators have many problems to translate technical terms. This is the reason why technical terms must be described and translated by developers, or best, if those terms will not be seen by end-users, then those should be kept in English, with a proper description.

On top of that, it is very hard for a non-developer to translate such an application before he/she knows how it works. By understanding the way ERP5 works, a translator will easily find the context of a term.

Skills

PO files should have been split into separate skill based PO files, assigned to skilled persons. Some persons are able to translate accounting, some are not. Some persons are able to translate developer terms, some are not, etc. Asking many different translators to translate the application can be a real gain of time, but only if those translators are skilled into what they have been assigned to translate, or assisted enough by skilled persons in order to learn. This learning procedure will take more time but will be useful because a new person will in the end be considered as skilled.

Working on the .po files

This chapter explains how to merge the two .po files and how to translate the terms of a .po file with Kbabel. Once all terms are translated, the first version of the localisation becomes available.

Merging .po files

This paragraph explains how to merge .po files. You will have indeed to merge the two .po files that you extracted from the Glossary and from ERP5, in order to create a unique .po file and process it. In order to merge several .po files, follow this

procedure:

Merging .po Files example before merging

.po file 1	.po file 2
# ERP5 Translation. #, fuzzy msgid "" msgstr "" "Project-Id-Version: PACKAGE VERSION\n" "PO-Revision-Date: 2008-03-03 17:13+0100\n" "Last-Translator: FULL NAME"	# ERP5 Translation. #, fuzzy msgid "" msgstr "" "Project-Id-Version: PACKAGE VERSION\n" "PO-Revision-Date: 2008-03-03 17:13+0100\n" "Last-Translator: FULL NAME"

The top area is the header entry of .po file. Every .po file requires this header information, but this is not a translation. The lower area is the translation area.

To merge these .po files, copy the lower area of the .po file 2 and paste it at the end of the blue area of .po file 1. The result is displayed in the table below: The .po files have been merged as described above.

Merging .po Files example after merging

.po file 1	.po file 2
# ERP5 Translation. #, fuzzy msgid "" msgstr "" "Project-Id-Version: PACKAGE VERSION\n" "PO-Revision-Date: 2008-03-03 17:13+0100\n" "Last-Translator: FULL NAME"	

Then, use msguniq command to unify duplicate translations, as described bellow:

```
$ msguniq -o unified_translation.po translation.po
```

The -o option is used to specify the output file. The result of this command line is a file names unified_translation.po and described bellow: unified_translation.po

Merging .po Files example after merging

unified_translation.po
ERP5 Translation. #, fuzzy msgid "" msgstr "" "Project-Id-Version: PACKAGE VERSION\n" "PO-Revision-Date: 2008-03-03 17:13+0100\n" "Last-Translator: FULL NAME"

The lower part contains the well merged entries. The highlighted section labelled "fuzzy" denotes a conflict entry, because there were two different translations in your .po files. In such a case, msguniq leaves both translations and set fuzzy mark on it. So you can find fuzzy terms and fix them later. You can also use msgcat command to do this in one step.

About KBabel

In order to process .po files, we would recommend to use KBabel [\[RD\]](#). Once KBabel is installed, you can read its documentation in order to understand how this application works. To read KBabel documentation, launch KBabel and open the Help tab.

A new application, named Lokalize [\[RD\]](#) is now superciding KBabel and should also be suitable for translating ERP5. We will update the documentation in the future to take into account Lokalize.

Alternate Solution

If you are not well versed in using command lines, an alternate solution consist of:

- Registering the .po file extracted from the Glossary (local terms) into the translation memory of KBabel,
- working with KBabel on the .pot file extracted from ERP5 (notification messages) and requesting KBabel to suggest translations from its translation memory.

This section of the documentation will be extended in the future in order to explain how KBabel and Lokalize built-in utilities can be used to increase productivity. Contributions are welcome.

Known Issues

It is not possible to translate the same English term to multiple different terms in different languages. To eliminate this issue, defining non ambiguous terms in the English Glossary is the best solution.

Committing a Translation

This chapter explains how to commit the .po file in order for the translation to be taken into account by ERP5.

Localisation Business Template and committing .po file

If you have a commit right on erp5 repository, you can commit your translated .po file. If you do not have a commit right on erp5 repository, please write to erp5-dev mailing list and you will be provided a commit right on the translation repository. New translators are very much welcome.

There is a naming rule for localisation business template. The prefix of id is "erp5_l10n_". suffix is either the language code or the language code and country code. For instance,

- erp5_l10n_ja, "ja" is a language code for Japanese.
- erp5_l10n_pt-BR, "pt" is a language code for Portuguese and "BR" is a country code for Brazil.

If localisation business template already exists, then it is easy to commit the .po file. In order to do this, check out the business template from the repository, replace the .po file with the new one, then commit it. This procedure prevents the new translation from being mixed with the ancient one, it replaces it.

For example, on the Japanese localisation business template, the .po file for erp5_ui is located here:
erp5_l10n_ja/MessageTranslationTemplateItem/ja/erp5_ui/translation.po. You can override translation.po with your new PO file.

If there is no localisation business template for your language yet, you have to create a business template for your local language, following this procedure:

- Add your local language in the Localizer or your ERP5 instance,
- Create an empty business template,
- Set your language in the Message Translations field of the business template,
- Commit the business template.

Although it is possible to commit this business template with erp5_subversion, it is a good idea to manually clean up the generated po file as described in the next paragraph.

Normalizing .po files to minimize changes

In order to make review of changes made in the .po file easier, developers are encouraged to clean up the .po file before committing it to svn. The msgattrib command (from gettext package) can be used for that purpose:

- Detect syntax errors in the .po file
- Remove fuzzy translations
- Remove non translated messages
- Sort messages alphabetically
- Don't wrap messages on more than one line

This can be done by this command:

```
msgattrib translation.po --no-fuzzy --translated -s --no-wrap -o translation.po
```

Cleaning up po files with this command before checking in guarantees a minimum difference between revisions and makes review easier.

Finalizing Translation with Localizer

This chapter explains how to use the Localizer properly. Using the Localizer is recommended because it helps working interactively on .po files and to watch in real time the results of translation in an ERP5 instance.

Test ERP5 Instance

The localizer helps you translating the terms of ERP5 interactively.

In order to do an interactive translation with the Localizer, you will need both a test ERP5 instance and an access to the Localizer on that instance. If you need a test ERP5 instance and you do not know how to set it up yourself, Nexedi can provide you one for free for translation purpose. Get in touch via [the contact section](#).

If you know how to setup an instance, then setup an instance and import the business template which you created at the previous step.

Translating with Localizer

Access the localizer with a first Web browser (ex. Konqueror). Go to the test instance you will work on. Log-in with the admin account (a.k.a. Manager account in Zope terminology), in order to be able to access the localizer. Use the following page: https://www.myerp5.com/YOURSITEID/manage_main, and click on the Localizer.

Launch another browser (ex. Chrome) and access <https://www.myerp5.com/YOURSITEID/view>. This is the address of your ERP5 test instance. When on your ERP5 test instance, login as normal user (not Manager). Once you are logged in, browse your instance. Whenever you find an untranslated word, translate it in the localizer (Konqueror in our example). While you translate, you must keep in mind that there are two distinct parts of the Localizer: the **Content (_content)** and the **User Interface (_ui)**.

- **Content:** this section gathers all (almost all) the terms that are located in rolling menus of your instance. If you want to translate them, click on "erp5_content (ERP5 Localized content)".
- **User Interface:** this section gathers everything that is not part of the content, eg words that are not in rolling menus. If you want to translate them, click on "erp5_ui (ERP5 Localized interface)".

Messages Propriétés Importer/Exporter TMX Langues Annuler Propriétaire Sécurité

MessageCatalog at /Localizer/erp5_content Help!

Accounting

Only untranslated messages

Résultats 1-15 de 901 >>

- Accounting
- Accounting & Finance
- Accounting & Finance Manager
- Accounting Agent
- Accounting Manager
- Accounts Payable
- Activities of membership organizations
- Activities of private households
- Administration
- Administrative Support
- Advertisement
- Afghan
- Afghanistan
- Agent
- Agriculture

Note:

English Français

Comptabilité

Save Effacer

Notes:

- Ensure a language is selected. On the screenshot, **French** is selected. You will enter the translation here.
- Above this field, the original term is displayed. Do not edit this field.
- Use the search zone to find the word you want to translate (case sensitive).
- Below the search you can see the search results. Click on the term you want to translate and after translating it, click on "Save"

Once a term is translated, you can refresh (F5) your Test Instance in your browser to see how it looks like.

ERP5 Translator Mode

Another way to translate terms in ERP5 is to activate the translator mode in Nexedi ERP5. This mode is only available to admin users ("Manager" role in Zope terminology). In this mode, every field of every form in ERP5 display 2 flags. Clicking on the flags brings directly to the Localizer, either to translate the field caption or the field description. This mode is useful to accelerate the initial work of finalization. However, it does not cover all cases, and in particular does not replace the procedure described at the previous section.

Tips

There are the two problems which you might face while trying to see the results of the translation in your test instance: Cache Problem and Script Problem.

The **Cache Problem** is related to the fact that certain parts of the user interface of ERP5 are calculated and stored in a cache memory. As a result, changing the translation terms in Localizer does not have an immediate effect. In order to solve

the cache problem, it is recommended to open the portal_caches in the same browser as the localizer. In order to do so, you must access the URL https://www.myperp5.com/YOURSITEID/portal_caches/manage and login as admin (Manager in Zope terminology). Once you access this page, click on "Clear all cache factories" to clear the cache of your instance. You can now go back to Firefox and refresh your page again, the term should now be translated. (If it doesn't, check that you selected the appropriate language in the context menu bar, or read on).

Translation scripts in ERP5 are still not perfect, which means that some modules do not apply the translations of the localizer. Therefore a term can be translated in a given module, but not in an other module. This is not your fault, and you can do nothing about this. For now, we are sure that at least the "Product" and "Documents" modules do not work fine as far as translation is concerned. Those bugs should be fixed soon.

Bugs

During the translation process, it is possible that some messages shown by the application are not translated or not translated correctly, in some cases the translation is not automatic and developers have to explicitly call "translateString" function to translate messages, sometimes they forget and message is not translated. If you find such a problem, please report it as a bug.

Another common problem is a wrong use of the translation that would cause an infinite number of messages to be generated. For example, translated messages could be:

- "1 object has been deleted"
- "2 objects has been deleted"
- "3 objects has been deleted"

This kind of bug happens when developers forget to state that certain items should not be translated. If you find such a bug, please report it.

Alternate procedure: import .po file into Localizer

Rather than committing the .po file and using a business template as a way to import a translation into a test instance as described in the previous chapter, some people prefer to import the .po file directly into Localizer and finalize it first. This is an acceptable approach which implies exchanging the two last steps of the procedure (this chapter and previous chapter). However, Localizer import and export features have proven to be sometimes very sensitive. This is the reason why we recommend the current approach.

Shortcomings and Future Directions

This documentation introduces the concept of Glossary and explains why it is essential to produce good translations of ERP5. This is its main contribution. It highlights the necessity to finalize a rough translation made through .po files by using interactive translation tools. It also provides a procedure which has proven to work in order to produce an acceptable French translation of ERP5 Express, after trying other procedures which have failed.

This documentation still had some shortcomings:

- it is too complex for beginners;
- it requires to use .po normalization procedures which, in our opinion, should be built-into the business template class;
- it does not explain well why we have to translate some messages in _content and why category titles are not part of _ui;
- it does not explain well how to commit changes which have been made to the translation using Localizer and how the normalization process operates in that case.

The next version of the documentation will provide answers to these issues.

`${related_subject_list}`