Translating the Tiki interface

This page is a kind of tutorial, that needs needs review. Moreover, some more screenshots would be nice... See related page: i18n Admin.

See also * Interactive Translation of Tiki Interface

1.1. Status of the translations
See: http://i18n.tiki.org/Status

1.2. Translate through i18n.tiki.org
One of the easiest ways to contribute to the translations of the Tiki interface is registering at i18n.tiki.org and use the interactive translation feature to get most of the strings available translated into your known languages.

That usually means committing changes to the next development version of Tiki (since these changes will be merged into the trunk branch of development).

See: http://i18n.tiki.org

1.3. Translate editing the language.* files on disk and use svn

For example, to improve the translation into Catalan, it is necessary to modify the content of the file:

./lang/ca/language.php

And since tiki6.1, you can edit the strings at:

./lang/ca/language.js

These .php (and .js) file contains a bunch of lines with pieces of sentences or words ("strings of text") like:

```
Code

"Internationalization" => "Internacionalització (I18n)",
"I18n setup" => "Configuració de I18n ",
```
"Best Language" => "Millor idioma",
"Restrict available languages" => "Restringeix idiomes disponibles",
"Available languages:" => "Idiomes disponibles:",
// "Intertiki exchange feature" => "Intertiki exchange feature",
// "Intertiki client" => "Intertiki client",
(...)

Lines that start with "// ", such as:

```php
// "Intertiki exchange feature" => "Intertiki exchange feature",
```

have not been translated yet. It is necessary to translate the right side of the line (to the right of "=>", the full text between the right set of double quotation marks (e.g. "Intertiki exchange feature" in the example above). Then delete the two initial slashes and the blank which follows ("// "). This way, the previous example would be (once translated):

```php
"Intertiki exchange feature" => "Funcionalitat d'intercanvi Intertiki",
```

Once you have translated a few strings, save the file language.php (and return a saved copy back into the original place you got it, if you had moved it somewhere else to edit it. Then execute the command:

```bash
http://yourdomain/get_strings.php?lang=ca
```

in order to ensure that Tiki understands your changes, and re-sort the text strings grouping the ones that still need to be translated, etc.

You can then continue translating, etc., and repeating the process as frequently as desired.

In the end, the file will need to be uploaded to the Tiki SVN (the central repository of code; keep reading, further down).

This general process has some variants, depending on how to edit the file that contains the text strings. Shortly, you can modify:

1. Each time directly against the central repository svn. How to commit
2. Previously through Wiki pages
3. Previously through Tiki itself (database)

Be careful that the language.php is a php parseable file. So you must not change some lines at the top

```php
<?php // -*- coding:utf-8 -*-
$lang=Array(
```

And some lines at the bottom
Be also careful to respect the syntax on each line

```
"..." => "...",
```

Each string is surrounded with double quotes ". If you need to add a double quote in your string, precede it with a \ like this : ". Do not forget the comma at the end of each line. Separate the 2 strings by =>.

### 1.3.1. Character sets

The file must be saved in utf8 encoding. 
If you're using Emacs (external link) then it's easy to change the encoding of a file. Simply open the file, 
and then type '-+C-x RET f+-' which runs the commandset-buffer-file-coding-system. Now choose utf-8 from the list.

### 1.3.2. How to contribute your translation improvements to Tiki

In order to have your translation improvements contributed to Tiki code (so that they are included in new releases, etc), you have to commit then to the branch equivalent to your tiki installation. Or to trunk branch.

For more information about branches and how to commit, check:
http://dev.tiki.org/Commit

For more information on how to merge strings translated from trunk into previous branches, see:
http://dev.tiki.org/Merge+language+improvements

### 1.3.3. Reference sites

- OpenTrad Apertium: On-line free software translator, including Catalan-French, French-Catalan, Catalan-English, English-Catalan, among others.
  http://xixona.dlsi.ua.es/prototype/
  http://www.opentrad.org

- Dictionary English-Catalan (open source code):
  http://www.ibiblio.org/dacco/cat/

### 1.3.4. PHP/Smarty translations

#### 1.3.4.1. Others: Convert to .PO file

Also there is the option to convert the language.php to a .po file, in order to take advantage of other tools which support the translation of PO files using the GNU Gettext.

This system is not explained here. You can find more information at:
1.3.4.2. Translation through Tiki interface (database)

You need to enable the option: "Admin > i18n (Internationalization) > Use database for translation", and afterwards, clean caches ("Admin > System Administration" > ./templates_c/ Empty).

Then, the link is enabled in the main application menu: Admin > Edit languages, that will bring to an url like:

http://yourdomain/tiki-edit_languages.php

Then it is necessary to create the translation into the database, for the Catalan language in this example, and tell Tiki to import the text strings already translated at the file language.php. This is performed in some short steps:

1. Tell Tiki to create a record for the new language (Catalan) into the database
   1. Create Language
      Shortname: ca (like en)
      Longname: Catalan (like English)
   2. Click in the Create button

2. Select the language you want to edit (Catalan) in the drop down menu.
   1. Select the language to edit: ca
   2. Click on the button Set

3. Then go to the link above called: Im- Export Languages,
   that will lead you to to an url like:
   ◦ http://yourdomain/tiki-imexport_languages.php?

4. Import the text strings of the language.php in Catalan
   1. Go to "Import > Select the language to", and choose Catalan (Catalan,ca) in the drop down menu. # # Click on the button Import.

5. Clean cache ("Admin > System Admin" > empty the three sections).
   You will see already the interface in your new language (Catalan, in this example).

To edit the text strings, then you need to:
Go to "Admin > Edit languages". In "Edit or export/import Languages", select the language you want to edit (do as explained above), and afterwards go to "edit Translations".
Then you will be able to edit the text strings you want, search any of them, etc. The process is quite intuitive.
Once finished, export the translation to the file language.php (through the link "Import/Export Languages"), and select the language you want to export. It is necessary that the file and unix folder are writable for the apache user, if in a GNU/Linux or similar environment.

At the end, it is necessary to take this generated file language.php (in ./lang/ca/language.php) and upload it (commit changes) the central repository of code of Tiki at sourceforge.net, through SVN, in case you want that other people can benefit from your improvements (desirable! 😊).

An example of adding a language

1.3.5. Custom translations

It is possible to customize your strings translation without changing the lang/your_language/language.php file. It is very convenient when you upgrade or sync with the latest. You can keep your translations. Your strings must be put in a file lang/your_language/custom.php. This file must be readable by the server. The file must have UTF-8 encoding. The format is very close to the format of language.php
<?php
$lang_custom = array(  
    "Hello"=>"My Hello",
    "This is the place to customize your translation"=>":-)",
);
$lang = array_merge($lang, $lang_custom); //###trebly: this changed will not be probably necessary for 7.x after 01/18/2011

In Tiki 9.x there is this on the last line:

$lang = $lang_custom + $lang;

A file example is located at lang/fr/custom.php_example.
You can customize any translation with this file by putting exactly the same string on the left as in language.php. The strings that are not in custom.php are still translated as specified in language.php. You can even change the English strings by using this file. For instance, if you do not like 'Contact Us' but prefer 'Contact Form' you can have a lang/en/custom.php like this

<?php
$lang_custom = array(  
    "Contact Us"=>'Contact Form",
);
$lang = $lang_custom + $lang;

Note: Be aware that any file like "language_*.php" will be also included in the translation.

When you are on a multitiki, you can have a lang/your_language/custom.php and a lang/your_language/your_tikidomain/custom.php

1.3.5.1. Custom JavaScript Translations
Some text can only be translated with JavaScript in the client browser. To do this add a file lang/your_language/custom.php using code in this format (as found in lang/fr/custom.js_example)

lang = $.extend(lang, {
    "Favorite" : "My favourite!",
    "Example in English" : "Translated text here"
    // remember IE does not support ending comma on last item
});

Be sure the file is saved with UTF-8 encoding. If the file is not saved with UTF-8 encoding, a fatal error may result and no Tiki pages will display. Terms/phrases in the custom file will be used in place of the same terms/phrases in the regular language file. (Not just anything can be added to these files, of course. They must be strings specified for translation in the templates, etc.)

1.3.5.2. Custom JavaScript Translations Without a File
If you cannot access the file system you can add similar code into the Custom JavaScript preference on the Look & Feel control panel.

if (jqueryTiki.language === "fr") {
1.4. More Information

- Internationalization
- Internationalization Admin
- Interactive Translation of Tiki Interface
- Multilingual Wiki
- http://tiki.org/i18n

Alias names for this page

Translating Tiki interface | Translating interface