

## MultilingualStep1

As some people were curious about the lang field I added in the page table, there are some explanations(I asked Mose the same question when he added the lang field in article)  
Feel free to comment ....

The purpose of this is to begin to work on the multilingual content. A real good multilingual system is hard to develop (see [Marc Laporte page](#)): It must include things like a translation workflow ....

The **first step** will deal only with the possibility to link together pages/articles of different languages. In consequence, when the detail of an object is displayed, its language is displayed with the languages list of the set of translations it belongs to.

1. a new language column for each object
  2. a table (object\_type, object\_id, translation\_id, language)
  3. a new page, like the categ page, where a user can update the translation link between the objects (tiki-edit\_translation.php)
- A user can update the language field in the edit page of the object or on the translation page.
  - An object can be in a translation set with no defined language (ex: a bilingual page can be translated in only one language). *1.9RC1 : an object must have a language to be in a translation set, because I don't know how to identify the object in the list of available language*
  - In a translation set , there can be 2 objects for the same language (this will be make sense with versionning). *1.9RC1 : this is not possible, because I don't know how to identify the object in the list of available language*

This principle can be applied also for other object : ex comments (object comments / forums posts)

The **step 2**: 2 pages in different languages can have the same name (this doesn't apply for article as the id is a number)

A contribution that will help :

*Maria Kretchetova wrote: Namespace conflicts occur when different users choose the same name for different pages. We introduce suffixes in order to get around this problem. A suffix consists of an underscore followed by word characters at the end of a page name. The key idea is that new pages inherit the suffix of the page where they were first referenced. This is implemented with an edit handler.*

The **step 3** is to organize the choice of the language the object is displayed with.

- (l) an object can be displayed in a specific language;
  - (?) an object can be displayed in the first language that matches
  - the user language,
  - eventually the language you come from,
  - the browser language preferences,
  - a translation quality,
- (in step 4 perhaps, we can add a user languages preferences, the browser preferences is not always set)

An important point to this approach:

- the links in a page don't need to link to the page in the appropriate language, the automatic language selector will find the page in the best language

....

Some disturbing points:

- an object has its own comments, a user doesn't know that one of the translated object has other

comments.

It is only a beginning....

### **Some other devs**

MikeW developped (at the same time) a multilingual system for the wiki pages.

It is like the first step described above except he chose some different functionalities:

- There is a "major / parent" page that is the first page of a set of translations. The major page is in the site default language. The first page is necessary to have other translations
- the list of available languages is displayed in a top bar, (flag, languageName)