
Subject: Re: Index page translation

Posted by [dolik.rce](#) on Thu, 06 May 2010 12:07:43 GMT

[View Forum Message](#) <> [Reply to Message](#)

andrei_natanael wrote on Thu, 06 May 2010 08:43 Another concern i have is about page layout. If one change the layout of one page the change will not propagate to other pages (e.g. translated pages) because the layout is included with every page so Topic++ doesn't help too much here. What if we have one page (for every page) with text in english and a tool which will extract the text from it, put it on .t files. The translators will translate text from .t file and the tool will integrate the text back in page for every language, in case that the layout of the page has changed the change will be visible to all pages. It also help in case that there are some modifications to original text.

For example:

We have index\$en-us.tpp. A tool will parse the file ignoring T++ tokens and extract all text in one file named index.t. The translators will do their job and at midnight when site get synced the tool will integrate translations in page so if we have in index.t file translations for Spanish and French the tool will generate index\$es-es.html and index\$fr-fr.html using the layout from index\$en-us.tpp file so if any change appear to main index it will propagate to other translated indexes without the need of translator to modify by hand the layout of page.

Andrei

Hi,

I agree that having single layout would be very helpful. But I'm not sure if translating it with .t files is the best option as it would force translators to switch often between the .tpp and .t to see the context and that would be not very comfortable.

Better approach IMHO would be to change the T++ system a bit. The main idea would be having a single .tpp file for each topic and a DropList, to select which language should be displayed/edited. I believe that this would also help in most other situations. In 99% of cases, the documentation for each language is almost identical, with same layout, so even regular users would benefit from this solution. The underlying implementation could be .t file and some QTF equivalent of t_ macro in .tpp that would be evaluated when interpreting the QTF. This solution would be a bit more difficult to implement and would have impact on whole U++, not just uppweb. Also care about backward compatibility would have to be taken.

What do you think?

Honza
