

## Bugs & Wish list

Construction error in \$lang, \$lang\_custom ?? | Tiki Wiki CMS Groupware :: Development  
Construction error in \$lang, \$lang\_custom ??

### Status

✖ Closed

### Subject

Construction error in \$lang, \$lang\_custom ??

### Version

18.x

### Category

- Support request

### Feature

i18n (Multilingual, l10n, Babelfish)

### Resolution status

New

### Submitted by

hman

### Lastmod by

hman

### Rating

★★★★★ (1) ?

### Related-to

- ✖ Missing "array\_merge" when the custom.php file is created by Tiki (admin, I18n)

### Description

Today I have fought with a stubborn error in the output of my tools that extract and process the contents of tra() clauses. I could not find out what on earth changed key/value pairs, until I found an astonishing explanation: It's a bug, no, actually a feature of PHP that I didn't think of, and I believe also the devs who constructed Tikis tra() logic.

Basically there is an include file language.php which defines something like this:



```
$lang = [ "first key" => "first value", ... "nth key" => "nth value", ];
```

Then there is another include file, custom.php, which can be edited through the custom translation admin page, which does something like this:



```
$lang_custom = array( "something" => "else", );
```

Then -if the bug ever gets fixed that this is omitted- custom.php close with a merge:



```
$lang = array_merge($lang, $lang_custom);
```

This is used so users/admins can modify the translation given in language.php. Newer values for keys given overwrite older definitions upon an array\_merge.

Normally, this works as intended. BUT something completely different can happen if keys with numbers appear...

Voilà, according to PHP manual

"If, however, the arrays contain numeric keys, the later value will not overwrite the original value, but will be appended. "

Which produces very funny (NOT) results, which are very hard to debug. And - language.php for German (de.de) DOES contain several numeric keys.

Also, a user or admin could at any time enter numeric keys. IMHO this "feature" should be documented and measures against numerical user entries be taken, unless I am completely mistaken. But simply try this little PHP demonstrator:

\$lang is taken from the real language.php.



```
<?php $lang = [ "1" => "1", "5" => "5", "10" => "10", "15" => "15", "30" => "30", "2" => "2",  
"4" => "4", "6" => "6", "7" => "7", "8" => "8", "9" => "9", "20" => "20", "40" => "40", "50" =>  
"50", "60" => "60", ]; $lang_custom = array( "something" => "else", ); $merge =  
array_merge($lang, $lang_custom); print_r($merge); ?>
```

#### Solution

<https://gitlab.com/tikiwiki/tiki/-/commit/682ab40b7b0de60853b2839a59986b0d1279170a>

Backported in Tiki 23.x, 22.x, 21.x, 18.x

#### Importance

8

#### Easy to solve?

5

#### Priority

40

#### Demonstrate Bug (Tiki 19+)

Please demonstrate your bug on [show2.tiki.org](https://show2.tiki.org)

Version: trunk ▼

#### Demonstrate Bug (older Tiki versions)

#### Ticket ID

7712

#### Created

Saturday 27 March, 2021 17:16:29 GMT-0000

by hman

#### LastModif

## Comments



hman 27 Mar 21 17:30 GMT-0000

For those who don't have a local PHP handy, the output is astonishingly, but correctly (!)



```
Array ( [0] => 1 [1] => 5 [2] => 10 [3] => 15 [4] => 30 [5] => 2 [6] => 4 [7] => 6 [8] => 7 [9] => 8 [10] => 9 [11] => 20 [12] => 40 [13] => 50 [14] => 60 [something] => else )
```

And - you cannot completely get rid of numeric keys, because tra() gets called on raw numbers, like here:



```
// lib/wiki-plugins/wikiplugin_rr.php "0" => "## http.php:66", "1" => "## http.php:67", //
lib/prefs/morelikethis.php "1" => "## morelikethis.php:29", "2" => "## morelikethis.php:30", "3"
=> "## morelikethis.php:31", "4" => "## morelikethis.php:32", "5" => "## morelikethis.php:33",
"6" => "## morelikethis.php:34", "7" => "## morelikethis.php:35", "8" => "##
morelikethis.php:36", "9" => "## morelikethis.php:37", "10" => "## morelikethis.php:38", //
lib/prefs/calendar.php "1" => "## calendar.php:67", "5" => "## calendar.php:68", "10" => "##
calendar.php:69", "15" => "## calendar.php:70", "30" => "## calendar.php:71",
```



hman 27 Mar 21 17:55 GMT-0000

One comment in the manual also provides a possible solution: Don't use array\_merge, but the union operator:



In some situations, the union operator ( + ) might be more useful to you than array\_merge. The array\_merge function does not preserve numeric key values. If you need to preserve the numeric keys, then using + will do that.

With that, you get

Array

```
(
1 => 1
5 => 5
10 => 10
15 => 15
30 => 30
2 => 2
4 => 4
```

```
6 => 6
7 => 7
8 => 8
9 => 9
20 => 20
40 => 40
50 => 50
60 => 60
something => else
)
```



hman 27 Mar 21 19:17 GMT-0000

Hm, the union operator does not handle re-defined values the same way as `array_merge` (Tiki needs this for corrections on translations), so the union operator is no easy replacement for `array_merge`. I guess a complete, new, subroutine must be written.



hman 27 Mar 21 22:03 GMT-0000

A little googling gives answers to a possible, but not desirable solution: union operator with reversed `$lang_custom` and `$lang` (not desirable because it disturbs the order given in `$lang`), but a "real" replacement could be `array_replace...`



```
<?php $lang = [ "1" => "1", "5" => "5", "10" => "10", "15" => "15", "30" => "30", "2" =>
"2", "4" => "4", "6" => "6", "7" => "7", "8" => "8", "9" => "9", "20" => "20", "40" => "40",
"50" => "50", "60" => "60", ]; $lang_custom = array( "something" => "else", "1" => "1b", );
$merge = array_merge($lang, $lang_custom); print_r($merge); $merge2 = $lang +
$lang_custom; print_r($merge2); $merge3 = $lang_custom + $lang; print_r($merge3);
$merge4 = array_replace($lang_custom,$lang); print_r($merge4); $merge5 =
array_replace($lang,$lang_custom); print_r($merge5); ?>
```

The fifth result looks fine (until we find another side effect 😊)



Jean-Marc Libs 30 Mar 21 13:56 GMT-0000

According to <https://www.php.net/manual/en/function.array-replace.php>



```
$merge5 = array_replace($lang,$lang_custom);
```

Is precisely what we need.

Also, should we encourage someone to remove all translations of numbers? This opportunity seems to only have ever been used twice in tiki18:

"1" in lang/ko/language.php

and "7" in lang/hu/language.php



hman>Jean-Marc Libs 30 Mar 21 15:32 GMT-0000

Replied to

According... That is not enough! Numbers cannot be pumped through tra(). As long as that is done and there is a mention of those numbers in either language.php OR custom.php, array\_merge will disturb those numbers.



Jonny Bradley 30 Mar 21 11:21 GMT-0000

Are numbers ever actually translated? If not, why are they in the language file?

Either way, an ideal example for a merge request (and discussion) on gitlab perhaps? 😊



hman 30 Mar 21 15:29 GMT-0000

Yes! Look at my second comment to see where exactly. Lots of numbers are pumped through tra(). If you want to know why, I'm afraid you'll have to ask the devs who wrote that code.

And the problem is - the array\_merge interferes here! So even if you don't use the numeric keys in custom.php, even if you don't even have a custom.php, array\_merge will disturb those keys.

I bet you didn't have reports on this issue in all those years, did you? But it undeniably exists. Just run the demonstrator script and see for yourself... For every number of reports, one is and will always stay the first. Before that was entered, no one reported it, but that doesn't mean the issue did not exist before...



hman 30 Mar 21 15:40 GMT-0000

There are more issues with tra(). I mentioned before that tra() is blind to context, but to correct that is surely a "major endeavour". The other is that sometimes sentence punctuation marks are within the string that is passed through tra(), sometimes they're outside, which means you cannot change it

through translation.

This is especially bad for languages who, for instance, use different question marks. Spanish uses an "upside down", and it's in front of the sentence, and greek uses the same glyph we know as a semicolon as a question mark: [https://en.wikipedia.org/wiki/Question\\_mark#Greek\\_question\\_mark](https://en.wikipedia.org/wiki/Question_mark#Greek_question_mark).

Also in Arabic there is a different question mark.

And to make matters worse: My set of tools, that tell me the difference between the language.php I used so far (RC 5, a merge from the original Tiki 18.8 language.php and my custom.php with > 25,000 changes) and the tra() extract from Tiki source code tells me that I have around 2,000 strings that are VERY similar, but not identical: I will have to look at the code of tra() to understand why you can translate a sentence through language.php that does not contain the full stop that clearly is contained in the original function call... So tra will translate "Do this." when you give it "Do this" => "Translation". Oh my goodness. I have already spent man-months with a new German translation. And apparently I am the only one who tries a comprehensive translation at all, guessing by the size of other language.php...



Jonny Bradley 31 Mar 21 11:01 GMT-0000

We have plans for Translations Revamp do be started at the TikiFest Virtual 2021 hopefully!

#### Attachments

| filename | created | hits | comment | version | filetype |
|----------|---------|------|---------|---------|----------|
|----------|---------|------|---------|---------|----------|

No attachments for this item

The original document is available at [https://dev.tiki.org/item7712-Construction-error-in-lang-lang\\_custom](https://dev.tiki.org/item7712-Construction-error-in-lang-lang_custom)