
Subject: preg_replace operation distorts multibyte unicode characters

Posted by [kong](#) on Thu, 14 Jan 2016 00:11:16 GMT

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

When a \$where string includes multibyte characters such as Chinese, sometimes you will notice that the \$where clause is not interpreted correctly.

This problem can traced back to the operation `$where = trim(preg_replace("/s+/", " ", $where));` // replace tabs and newlines with ' ' in function `where2indexedArray ($where)` in file `include.library.inc`.

Turns out that the /s criteria of the regex applies also to individual bytes of multibyte characters and when such a single byte meets /s criteria it would then be replaced with a "space byte", changing the multibyte character into something else. I solved this problem by adding the /u modifier to the regex: `$where = trim(preg_replace("/s+/u", " ", $where));` // replace tabs and newlines with ' '

For reference: <http://www.regular-expressions.info/php.html>Quote:If you want your regex to treat Far East characters as individual characters, you'll either need to use the mb_ereg functions, or the preg functions with the /u modifier.

Subject: Re: preg_replace operation distorts multibyte unicode characters

Posted by [AJM](#) on Thu, 14 Jan 2016 09:53:50 GMT

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

Thanks for spotting that. I will include this change in the next release.
