

## No mbstrings attached: PHP UTF-8

It's interesting to see where setting Community Aims will get you. QS-E has a few so-called Community Aims, these are basically goals we must attain to support some of the most requested features in Quantum Star SE Evolved. A close second to the ideal of being able to transparently add new content to the game was the idea of allowing a full translation feature and native support for all multi-byte character encodings - We have gotten translation offers before for Hindi, Romanian, Russian, and more.

Recently given my prior blog post on a translation system I was looking at the system as a test implementation, and the broader area of I18N (Internationalisation). Further to my earlier post of today (Man, I write too much to this blog!) I found out from a [Planet-PHP.net](#) blog entry that [Harry Fuecks recently released a godsend](#). He's written and released the [0.1 version of a PHP UTF-8 library](#) which does NOT rely upon having the mbstring library available. This is fantastic news! Using this library one can actually operate on UTF-8 multi-byte characters safely - no corruption of multi-byte strings!

I modified a test included in the library download. It offers the following results - with my name finally represented accurately!

Operating on a 20 character string of both single and multi-byte characters...

The following represents String operations utilising a default PHP installation without the mbstring library.

String is: IÃfÃ±tÃfÃ«rnÃfÃçtiÃfÃ´nÃfÃ lizÃfÃ!tiÃfÃ,n

This is a UTF-8 string - standard PHP manipulations follow (no mbstring enabled):

Num chars: 27 (it's actually 20 - PHP cannot count characters since strlen() actually counts bytes)

Uppercase: IÃfÃ±TÃfÃ«RNÃfÃçTIÃfÃ´NÃfÃ LIZÃfÃ!TIÃfÃ,N (php ignored all the multi-byte characters)

Reversed:

nÃ-Ã¿Ã½Ã-Ã¿Ã½itÃ-Ã¿Ã½Ã-Ã¿Ã½zilÃ-Ã¿Ã½Ã-Ã¿Ã½nÃ-Ã¿Ã½Ã-Ã¿Ã½itÃ-Ã¿Ã½Ã-Ã¿Ã½nrÃ-Ã¿Ã½Ã-Ã¿Ã½tÃ-Ã¿Ã½Ã-Ã¿Ã½l (php corrupted the multi-byte characters, turning each into twin corrupted byte chars)

The following represents String operations utilising a default PHP installation without the mbstring library. The test does however utilise a PHP implementation of UTF-8 safe String functions.

String is: IÃfÃ±tÃfÃ«rnÃfÃçtiÃfÃ´nÃfÃ lizÃfÃ!tiÃfÃ,n

It's well formed UTF-8

Num chars: 20

Uppercase: IÃfÃ±TÃfÃ«¹RNÃfÃçŠTIÃfÃ•NÃfÃ,-LIZÃfÃ€ TIÃfÃËœN

Reversed: nÃfÃ,tiÃfÃ!zilÃfÃ nÃfÃ´itÃfÃçnrÃfÃ«tÃfÃ±l

My Name:

Maugrim's real name: PÃ¡draic

Reversed (Default PHP): ciardÃ-Ã¿Ã½Ã-Ã¿Ã½P

Reversed (with UTF-8 Lib): ciardÃ¡P

Posted by PÃ¡draic Brady in PHP General at 05:46