Subject: Running Asian Scripts in Radicore
Posted by nnones on Thu, 25 Jan 2007 11:26:17 GMT
View Forum Message <> Reply to Message

We are building an application using Radicore that needs to run in various East Asian languages as well as English; i.e. Thai and Simplified Chinese. I've run into a couple of issues and devised successful workarounds that I thought I'd share.

In function getLanguageText, the statement:

```
$string = convertEncoding($string, 'latin1', 'UTF-8');
```

does not work when the language_text.inc or sys.language_text.inc files are already encoded in another character set such as UTF-8. This would always be the case when the translator is keying in text in a language like Thai.

I studied the convertEncoding function and discovered that it automatically inserts 'UTF-8' when the second argument ('latin1' as called by the getLanguageText function) is null. So, the above statement should be:

```
$string = convertEncoding($string, '', 'UTF-8');
```

This seems to solve the problem. Of course it will only work if the text files are really saved in UTF-8 and not some other character set that's not latin1. To prevent the problem from recurring we have decided to encode all text files in UTF-8, except the ones that we are absolutely certain contain nothing but ASCII characters.

I have also discovered that when you are saving a text file using UTF-8 encoding, using the standard Notepad editor that ships with Windows, it saves the file as “UTF-8 with signature.” The signature is a special 2-byte character HEX(FEFF) that the Unicode standard recognizes as the “Byte Order Mark” (BOM). When Notepad saves the file, the BOM is inserted right at the beginning of the document. The resulting HTML pages that the Radicore XSLTs create, always contain the BOM unless the text file is saved in ASCII.

This wouldn't be a big deal except that Internet Explorer 7 has a bug that causes it to misbehave when it sees a BOM.

The solution is to save all the files containing non-ASCII text as plain old UTF-8, without the signature. But Notepad won’t let you do this. The “Notepad2” freeware text editor seems to work fine. It’s actually much nicer than Notepad so I plan to use it for all our work.

The third issue is really a question for Tony. Following the “Internationalisation and the Radicore Development Infrastructure” article, we enabled the multibyte string functions in our PHP (5.*) configuration. Not knowing exactly what to do, we decided to go all the way and implement all the overload functions in php.ini, as in:

```
mbstring.func_overload = 7
```

After this, I discovered that Radicore no longer processes multiple records when you select multiple rows from a screen such as LIST1. I have not investigated this problem any further but would like an opinion on whether function overloading is the culprit. I'd also appreciate a recommendation on what this setting should be.

Subject: Re: Running Asian Scripts in Radicore
Posted by AJM on Thu, 25 Jan 2007 15:53:49 GMT
View Forum Message <> Reply to Message

If changing $string = convertEncoding($string, 'latin1', 'UTF-8'); to $string = convertEncoding($string, '', 'UTF-8'); solves the problem then I shall implement that change in my code.

I think that if you are going to use East Asian languages then UTF-8 is the best encoding to use as it appears to cover most combinations. It is already used as the standard encoding for XML files.

I as not aware of the problem with saving files with Windows Notepad, so it may be worth a mention in my article. Do you know if the same problem exists with Wordpad?

As for problems with the mbstring.func_overload option, I have never tried this so I have not encountered any errors. If some problems arise when it is turned on then it would be useful to investigate further to see exactly what is happening. I would be grateful for any information that you could provide.

Subject: Re: Running Asian Scripts in Radicore
Posted by nnones on Thu, 25 Jan 2007 16:46:50 GMT
View Forum Message <> Reply to Message

To the best of my knowledge, WordPad is not able to save files in UTF-8 format. It does have a "Unicode Text Document" option for saving files but I think this is 2-byte (UCS-2) Unicode, not multi-byte (UTF-8). It is therefore inappropriate for Thai characters, which occupy 3 bytes in UTF-8 format. It probably wouldn't work properly for other character sets, either, even those that take up 2 bytes in UTF-8.

If you change the Radicore code to eliminate the conversion from latin1, then I suggest that you amend your article to make UTF-8 encoding mandatory for all language_text.inc and sys.language_text.inc files that contain non-ASCII characters. This is because the convertEncoding function would no longer convert non-UTF-8 files accurately -- for example, it would no longer handle latin1 files having characters outside the ASCII character set.

The article should also recommend using UTF-8 encoding everywhere as a best practice, for it produces exactly the same output as ASCII encoding (for ASCII characters) whilst handling virtually every other language on Earth.

I will report what I learn as I test the various mbstring.func_overload options.

Subject: Re: Running Asian Scripts in Radicore
Posted by nnones on Wed, 07 Mar 2007 16:17:32 GMT
View Forum Message <> Reply to Message

I have completed some testing of the mbstring.func_overload options.

Earlier, we had set the mbstring.func_overload option to 7, which implements all 3 overload function categories: mail, string and regular expression functions. We found that although this setting handles UTF-8 strings correctly, Radicore no longer processes multiple records when you select multiple rows from a screen such as LIST1.

We have learnt that when you set the mbstring.func_overload option to 2 (string overload functions only), the application handles UTF-8 strings correctly and Radicore is able to process multiple records when you select multiple rows.

When we set the mbstring.func_overload option to 6 (string and regular expression overload functions), Radicore no longer processes multiple records when you select multiple rows.

When we set the mbstring.func_overload option to 0 (no overload functions), our application does not handle UTF-8 strings correctly.

We did not test mbstring.func_overload option 1 (mail) since our application is not sending or receiving e-mail.

Therefore it appears that the right mbstring.func_overload option is 2 (string overload functions only).

Subject: Re: Running Asian Scripts in Radicore
Posted by AJM on Wed, 07 Mar 2007 18:01:52 GMT
View Forum Message <> Reply to Message

That you for that useful information.
