# Ruby - Feature #1784

## More encoding (Big5 series) support?

07/17/2009 12:17 AM - godfat (Lin Jen-Shin)

| | |
|---|---|
| **Status:** | Closed |
| **Priority:** | Normal |
| **Assignee:** | duerst (Martin Dürst) |
| **Target version:** | 2.0.0 |

**Description**

=begin
I was very glad to see there's build-in encoding support,
but if we could support more Big5 related encodings,
it would be much better, because there are many,
many Big5 extensions.

Current CP950 (from Microsoft) do not contain Japanese
nor Simplified Chinese, nor some Traditional Chinese characters.
Because of this, many Big5 extensions were invented.
The most popular Big5 extensions nowaday would be Big5-HKSCS and
UAO ( Unicode-at-on, http://uao.cpatch.org/ ).

libiconv supports Big5-HKSCS, but UAO not.
I am not sure about Big5 status in Honk Kong, but here in Taiwan,
the most used Big5 encoding was UAO. (I think)
For example, telnet://ptt.cc contains many, many Japanese
characters in UAO. It's a very popular BBS in Taiwan.

Here's a reference in Traditional Chinese from Mozilla Taiwan:
http://moztw.org/docs/big5/

There's `Mozilla 1.8' too, trying to merge some Big5
encodings into one, but I am not familiar with it.
At least I can use it to read most characters.

Here's the related issue from Mozilla:
https://bugzilla.mozilla.org/show_bug.cgi?id=310299

And here's the table they used:
Big5 to Unicode(codepoint):
http://moztw.org/docs/big5/table/moz18-b2u.txt
Unicode(codepoint) to Big5:
http://moztw.org/docs/big5/table/moz18-u2b.txt

I am trying to build this into Ruby, but I am
no expert in encoding nor Ruby core development.
The first experiment succeeded and I'm trying
to polish it later.

Could Ruby support more encodings in the future?
Or is there a way to add more encodings from
user library level?

Many Thanks!
=end

**Related issues:**

| | | |
|---|---|---|
| Related to Ruby - Feature #4073: HKSCS-2008 | **Closed** | **11/19/2010** |

### Associated revisions

**Revision e0436c54c21343580d5fa6b9334fbfa20e10c646 - 11/17/2009 08:56 AM - duerst (Martin Dürst)**

- enc/big5.c, enc/trans/big5.trans, enc/trans/big5-uao-tbl.rb,

test/ruby/test-transcode.rb: Added Encoding 'Big5-UAO' and transcoding
for it (from Tatsuya Mizuno) (see Bug #1784)

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@25822 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

**Revision e0436c54c21343580d5fa6b9334fbfa20e10c646 - 11/17/2009 08:56 AM - duerst (Martin Dürst)**

- enc/big5.c, enc/trans/big5.trans, enc/trans/big5-uao-tbl.rb,
test/ruby/test-transcode.rb: Added Encoding 'Big5-UAO' and transcoding
for it (from Tatsuya Mizuno) (see Bug #1784)

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@25822 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

**Revision e0436c54 - 11/17/2009 08:56 AM - duerst (Martin Dürst)**

- enc/big5.c, enc/trans/big5.trans, enc/trans/big5-uao-tbl.rb,
test/ruby/test-transcode.rb: Added Encoding 'Big5-UAO' and transcoding
for it (from Tatsuya Mizuno) (see Bug #1784)

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@25822 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

## History

**#1 - 07/17/2009 01:35 AM - naruse (Yui NARUSE)**

*- Status changed from Open to Assigned*

*- Assignee set to naruse (Yui NARUSE)*

=begin
OK, I'll implement it on Ruby 1.9.2 or 1.9.3.

But what do *you* want the variant of Big5?  UAO or Big5-HKSCS or both?

I'll implement encodings what people want to use, even if the encoding is unofficial.  But not want encodings what won't be used.

In current we don't provide a way to add user encodings.  Because we want feedbacks; what encodings are needed.
=end

**#2 - 07/17/2009 09:15 PM - godfat (Lin Jen-Shin)**

=begin
Very glad to hear so!

In short, what *I* would want would be UAO *or*
Big5 from Mozilla Taiwan (I would call it Moz18 later),
This is because Moz18's b2u table (Big5 to Unicode)
is the same with UAO 2.41's, so either one would be
fine in most cases. And perhaps I won't need u2b
transcoding :p

Here's the long reason and some background:
I know that there used to be many people installed
UAO to use Big5 Japanese in Taiwan. They were
taught that this software would complement (□□)
Unicode, and many people thought that UAO was
Unicode. This helped people using unified Big5 Japanese,
but slowed down the progress of going to Unicode.

As far as I know, UAO isn't any standard,
so there're few softwares support it,
except softwares that made for Taiwanese.
Most of them are telnet clients for BBS.

I am not sure but somehow Big5-HKSCS could
transcode many characters from UAO, and there
are many softwares support Big5-HKSCS, e.g. libiconv.
So using Big5-HKSCS could be a workaround if
UAO wasn't supported.

As for Moz18, the Big5 from Mozilla Taiwan,
I have to admit that I've never heard it before
reading this page: http://moztw.org/docs/big5/
It said what they have in mind were the most

compatible ability, and easing the problem of UAO.
(i.e. too few softwares support it)
Its b2u is the same with UAO 2.41 (but not 2.50), and
u2b is based on CP950, plus the extension part of
Big5-2003 and UAO.

Furthermore, they worked with the authors of UAO,
and UAO encouraged people to use Firefox to *read*
missing characters in Big5. So I would suppose this
would be the last yet another Big5 variant...

As a result, support for either Moz18 or UAO
would be fine enough for *me*.

Here's the tables for reference:
Moz18
In previous post.

UAO 2.41 (b2u is the same with Moz18's according to the page)
http://moztw.org/docs/big5/table/uao241-b2u.txt
http://moztw.org/docs/big5/table/uao241-u2b.txt

UAO 2.50 (I am not sure if this differs many from 2.41)
http://moztw.org/docs/big5/table/uao250-b2u.txt
http://moztw.org/docs/big5/table/uao250-u2b.txt

Sincerely,

=end

#### #3 - 07/18/2009 05:07 PM - duerst (Martin Dürst)

=begin
Hello Jen-Shin,

I have talked with Yui Naruse here at RubyKaigi 2009.
I will ask my student, Tatsuya Mizuno, to work on adding this encoding to
Ruby. I hope it will be finished within about one week.

Regards,  Martin.

On 2009/07/17 21:15, Lin Jen-Shin wrote:

> Issue #1784 has been updated by Lin Jen-Shin.
>
> Very glad to hear so!
>
> In short, what *I* would want would be UAO *or*
> Big5 from Mozilla Taiwan (I would call it Moz18 later),
> This is because Moz18's b2u table (Big5 to Unicode)
> is the same with UAO 2.41's, so either one would be
> fine in most cases. And perhaps I won't need u2b
> transcoding :p
>
> Here's the long reason and some background:
> I know that there used to be many people installed
> UAO to use Big5 Japanese in Taiwan. They were
> taught that this software would complement (□□)
> Unicode, and many people thought that UAO was
> Unicode. This helped people using unified Big5 Japanese,
> but slowed down the progress of going to Unicode.
>
> As far as I know, UAO isn't any standard,
> so there're few softwares support it,
> except softwares that made for Taiwanese.
> Most of them are telnet clients for BBS.
>
> I am not sure but somehow Big5-HKSCS could
> transcode many characters from UAO, and there
> are many softwares support Big5-HKSCS, e.g. libiconv.
> So using Big5-HKSCS could be a workaround if
> UAO wasn't supported.
>
> As for Moz18, the Big5 from Mozilla Taiwan,
> I have to admit that I've never heard it before

reading this page: http://moztw.org/docs/big5/
It said what they have in mind were the most
compatible ability, and easing the problem of UAO.
(i.e. too few softwares support it)
Its b2u is the same with UAO 2.41 (but not 2.50), and
u2b is based on CP950, plus the extension part of
Big5-2003 and UAO.

Furthermore, they worked with the authors of UAO,
and UAO encouraged people to use Firefox to *read*
missing characters in Big5. So I would suppose this
would be the last yet another Big5 variant...

As a result, support for either Moz18 or UAO
would be fine enough for *me*.

Here's the tables for reference:
Moz18
In previous post.

UAO 2.41 (b2u is the same with Moz18's according to the page)
http://moztw.org/docs/big5/table/uao241-b2u.txt
http://moztw.org/docs/big5/table/uao241-u2b.txt

UAO 2.50 (I am not sure if this differs many from 2.41)
http://moztw.org/docs/big5/table/uao250-b2u.txt
http://moztw.org/docs/big5/table/uao250-u2b.txt

Sincerely,

---

http://redmine.ruby-lang.org/issues/show/1784

---

http://redmine.ruby-lang.org

--
#-# Martin J. Dürst, Professor, Aoyama Gakuin University
#-# http://www.sw.it.aoyama.ac.jp mailto:duerst@it.aoyama.ac.jp

=end

#### #4 - 07/19/2009 02:47 AM - naruse (Yui NARUSE)

*- Assignee changed from naruse (Yui NARUSE) to duerst (Martin Dürst)*

=begin
Difference between 2.41 and 2.50 is following

```
% diff -su uao241-u2b.txt uao250-u2b.txt
Files uao241-u2b.txt and uao250-u2b.txt are identical
% diff -swu uao241-b2u.txt uao250-b2u.txt|less
--- uao241-b2u.txt      2005-09-30 03:34:20.000000000 +0900
+++ uao250-b2u.txt      2009-07-16 15:34:38.000000000 +0900
@@ -3611,16 +3611,16 @@
0x97FD 0xE931
0x97FE 0xE932
0x9840 0x9C76
-0x9841 0x278A
-0x9842 0x278B
-0x9843 0x278C
-0x9844 0x278D
-0x9845 0x278E
-0x9846 0x278F
-0x9847 0x2790
-0x9848 0x2791
-0x9849 0x2792
-0x984A 0x2793
+0x9841 0x2776
+0x9842 0x2777
+0x9843 0x2778
+0x9844 0x2779
+0x9845 0x277A
+0x9846 0x277B
```

+0x9847 0x277C
+0x9848 0x277D
+0x9849 0x277E
+0x984A 0x277F
0x984B 0x9C85
0x984C 0x9C86
0x984D 0x9C87

These 10 characters are "DINGBAT NEGATIVE CIRCLED SANS-SERIF DIGIT {ONE-TEN}" to "DINGBAT NEGATIVE CIRCLED DIGIT {ONE-TEN}".
=end

#### #5 - 07/20/2009 06:42 AM - naruse (Yui NARUSE)

=begin
Big5-UAO (UAO 2.50) seems suitable to bundle.
Big5-HKSCS seems also has worth.

Mozilla 1.8, its name may be Big5-moz, is still thinking.
Mozilla version of EUC-JP has the same problem.

When import table to Ruby, please import with its original table and conversion tools.

P.S. to Martin
How do you think about this?
"GB 18030: A mega-codepage"
http://www.ibm.com/developerworks/java/library/u-china.html en ver
http://www.ibm.com/developerworks/jp/java/library/u-china.html jp ver
=end

#### #6 - 07/24/2009 07:34 PM - duerst (Martin Dürst)

=begin
We added BIG5-HKSCS (2004 version; same as libiconv) transcoding today. Please check. If you have any tests (see test/ruby/test_transcode, at the very end) that you can add, please contribute them.

BIG5-HKSCS wasn't available as an encoding, so we added it as a replicate of BIG5 in enc/big5.c. However, this is technically not correct. The enc/big5.c file defines 0x8? and 0x9? as single-byte
ranges, but they are used as lead bytes for double-byte characters in BIG5-HKSCS (and other BIG5 variants, it seems).

I'd estimate that this feature request is currently about 30% done. How can I set the % Done field to 30%?
=end

#### #7 - 07/24/2009 10:37 PM - naruse (Yui NARUSE)

=begin

> We added BIG5-HKSCS (2004 version; same as libiconv)
> Thanks but please add original table with url and script for converting to *-tbl.rb.

> BIG5-HKSCS wasn't available as an encoding
> I'll add it.

> I'd estimate that this feature request is currently about 30% done. How can I set the % Done field to 30%?
> ignore it please.
> =end

#### #8 - 09/08/2009 11:25 PM - godfat (Lin Jen-Shin)

*- File test_big5-hkscs.rb added*

=begin
Hi, I've tested BIG5-HKSCS against trunk today, and it works fine.

# ruby 1.9.2dev (2009-09-08 trunk 24791) [i386-darwin9.8.0]

Many thanks for your effort. The attachment is the test code I used.
It can transcode successfully in first case, but not the other.
I think the failed one could be Big5-UAO encoded.

Thanks again.

=end

#### #9 - 10/11/2009 01:53 AM - naruse (Yui NARUSE)

=begin
Is Big5-YAO still working?
=end

#### #10 - 10/11/2009 03:12 PM - duerst (Martin Dürst)

=begin
On 2009/10/11 1:53, Yui NARUSE wrote:

> Issue #1784 has been updated by Yui NARUSE.
>
> Is Big5-YAO still working?

It is still being worked on. Please wait another week or two. Thanks.

---

> http://redmine.ruby-lang.org/issues/show/1784

---

> http://redmine.ruby-lang.org

--
#-# Martin J. Dürst, Professor, Aoyama Gakuin University
#-# http://www.sw.it.aoyama.ac.jp mailto:duerst@it.aoyama.ac.jp

=end

#### #11 - 11/17/2009 05:57 PM - duerst (Martin Dürst)

*- Status changed from Assigned to Closed*

*- % Done changed from 0 to 100*

=begin
This issue was solved with changeset r25822.
Lin, thank you for reporting this issue.
Your contribution to Ruby is greatly appreciated.
May Ruby be with you.

=end

### Files

| | | | |
|---|---|---|---|
| test_big5-hkscs.rb | 887 Bytes | 09/08/2009 | godfat (Lin Jen-Shin) |