

Ruby trunk - Feature #3418

IO#putc Clobbers Multi-byte Characters

06/10/2010 05:49 AM - runpaint (Run Paint Run Run)

Status:	Closed
Priority:	Normal
Assignee:	naruse (Yui NARUSE)
Target version:	2.0.0
Description	
<pre>=begin IO#putc claims to write a "character", when in fact it writes a byte. I assume this is for backward compatibility reasons, but as this could lead to data loss, the documentation needs clarifying. Currently, #putc doesn't require the stream to be in binmode, provide any warning of the truncation, or agree with IO#getc on the definition of "character". open('/tmp/putc', 'w+') { f f.puts "\u1234"; f.rewind; f.read} #=> "\xE1 open('/tmp/getc', 'w+'){ f f.print "\u1234"; f.rewind; f.getc} #=> "" If the IO stream explicitly specifies a non-BINARY encoding, the first example fails with an Encoding::UndefinedConversionError, which is reasonable. open('/tmp/putc', 'w+:UTF-8'){ f f.puts "\u1234"; f.rewind; f.read} #=> Encoding::UndefinedConversionError: "\xE1" from ASCII-8BIT to UTF-8 =end</pre>	

History

#1 - 06/10/2010 06:44 AM - matz (Yukihiko Matsumoto)

```
=begin
Hi,

In message "Re: [ruby-core:30697] [Bug #3418] IO#putc Clobbers Multi-byte Characters"
on Thu, 10 Jun 2010 05:49:55 +0900, Run Paint Run Run redmine@ruby-lang.org writes:

|IO#putc claims to write a "character", when in fact it writes a byte. I assume this is for backward compatibility reasons, but as this could lead to data loss, the documentation needs clarifying.

Agreed. The behavior is intentional, the term "character" in the documentation means a byte in 8bit ascii, not to apart from old putc(3) function in the C library. So this one is a documentation bug at most.
```

matz.

```
=end
```

#2 - 06/10/2010 07:15 AM - runpaint (Run Paint Run Run)

- File *io.c-putc.patch* added

```
=begin
Thanks. Patch attached.
=end
```

#3 - 06/10/2010 07:18 AM - runpaint (Run Paint Run Run)

- File *io.c-putc.patch* added

```
=begin
Drat. Wrong file; try this one.
=end
```

#4 - 06/10/2010 09:09 AM - matz (Yukihiko Matsumoto)

=begin
Hi,

In message "Re: [ruby-core:30701] [Bug #3418] IO#putc Clobbers Multi-byte Characters"
on Thu, 10 Jun 2010 07:18:58 +0900, Run Paint Run Run redmine@ruby-lang.org writes:

|File io.c-putc.patch added

Thank you for the patch. I will apply the patch, except for examples
for multi-byte characters, since I want to make it implementation
detail.

```
matz.
```

=end

#5 - 06/10/2010 09:17 AM - matz (Yukihiko Matsumoto)

- Status changed from Open to Closed

- % Done changed from 0 to 100

=begin
This issue was solved with changeset [r28243](#).
Run Paint, thank you for reporting this issue.
Your contribution to Ruby is greatly appreciated.
May Ruby be with you.

=end

#6 - 06/10/2010 11:41 PM - mame (Yusuke Endoh)

- Status changed from Closed to Open

- Assignee set to naruse (Yui NARUSE)

=begin
Hi,

I agree that this is an implementation detail, but I also expect IO#putc
to handle normal character, because IO#getc behaves so:

```
$ cat t.txt  
□□□□
```

```
$ ruby19 -e 'open("t.txt") {|f| p f.getc }'  
"□"
```

```
$ ruby19 -e 'open("t.txt", "w") {|f| f.putc ?□ }'
```

```
$ ruby19 -e 'open("t.txt") {|f| p f.read }'  
"\xE3"
```

IO#putbyte would be needed for the byte-oriented purpose.
I move this ticket to 1.9.x feature request.

--
Yusuke Endoh mame@tsg.ne.jp
=end

#7 - 09/14/2010 03:58 PM - shyouhei (Shyouhei Urabe)

- Status changed from Open to Assigned

=begin

=end

#8 - 10/12/2010 03:27 PM - naruse (Yui NARUSE)

- Status changed from Assigned to Closed

=begin

This issue was solved with changeset [r29447](#).

Run Paint, thank you for reporting this issue.

Your contribution to Ruby is greatly appreciated.

May Ruby be with you.

=end

Files

io.c-putc.patch	1.25 KB	06/10/2010	runpaint (Run Paint Run Run)
io.c-putc.patch	1.1 KB	06/10/2010	runpaint (Run Paint Run Run)