

Backport193 - Backport #5539

Readline.readline() blocks all threads

11/02/2011 03:39 AM - cjameshuff (Christopher Huff)

Status:	Closed
Priority:	Normal
Assignee:	
Description	
<p>The standard library <code>Readline.readline()</code> function blocks the thread it's called from without unlocking the VM, thus blocking all threads. This code demonstrates the problem:</p> <pre>require 'pp' t = Thread.new {10.times { x puts x; sleep 1}} while(buf = Readline.readline) p buf if(buf == 'q') break end end</pre> <p>Thread <code>t</code> will be blocked by the <code>readline</code> call, numbers being printed occasionally when lines are entered and <code>readline()</code> returns, rather than once per second. Reproduced on ruby 1.9.2p180 (2011-02-18 revision 30909) [x86_64-darwin10.6.0] and ruby 1.9.3p0 (2011-10-30 revision 33570) [x86_64-darwin10.8.0].</p> <p>The <code>RubyInline</code> gem allows a workaround:</p> <pre>require 'inline' class BetterReadline inline :C do builder builder.include " builder.include " src = <<END VALUE readline_intern(void * data) { char ** rstr = (char **)data; *rstr = readline(NULL); return Qnil; } END builder.prefix(src) src = <<END VALUE rb_readline(void) { char * str = NULL; rb_thread_blocking_region(readline_intern, &str, NULL, NULL); return rb_str_new2(str); } END builder.c_singleton(src, method_name: 'readline') end end # class BetterReadline</pre> <p>This then illustrates the desired behavior, with the thread running in the background while <code>readline()</code> waits for input:</p> <pre>require 'pp' t = Thread.new {10.times { x puts x; sleep 1}} while(buf = BetterReadline.readline) p buf if(buf == 'q') break end end</pre>	
Associated revisions	

Revision ec4f9d1c - 12/23/2011 07:02 AM - kouji (Kouji Takao)

- ext/readline/readline.c (Init_readline): libedit check rl_getc_function only when rl_initialize() is called, and using_history() call rl_initialize(). This assignment should be placed before using_history(). [ruby-core:40641] [Bug #5539]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@34108 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 34108 - 12/23/2011 07:02 AM - kouji (Kouji Takao)

- ext/readline/readline.c (Init_readline): libedit check rl_getc_function only when rl_initialize() is called, and using_history() call rl_initialize(). This assignment should be placed before using_history(). [ruby-core:40641] [Bug #5539]

Revision 34108 - 12/23/2011 07:02 AM - kouji (Kouji Takao)

- ext/readline/readline.c (Init_readline): libedit check rl_getc_function only when rl_initialize() is called, and using_history() call rl_initialize(). This assignment should be placed before using_history(). [ruby-core:40641] [Bug #5539]

Revision 34108 - 12/23/2011 07:02 AM - kouji (Kouji Takao)

- ext/readline/readline.c (Init_readline): libedit check rl_getc_function only when rl_initialize() is called, and using_history() call rl_initialize(). This assignment should be placed before using_history(). [ruby-core:40641] [Bug #5539]

Revision 34108 - 12/23/2011 07:02 AM - kouji (Kouji Takao)

- ext/readline/readline.c (Init_readline): libedit check rl_getc_function only when rl_initialize() is called, and using_history() call rl_initialize(). This assignment should be placed before using_history(). [ruby-core:40641] [Bug #5539]

Revision 34108 - 12/23/2011 07:02 AM - kouji (Kouji Takao)

- ext/readline/readline.c (Init_readline): libedit check rl_getc_function only when rl_initialize() is called, and using_history() call rl_initialize(). This assignment should be placed before using_history(). [ruby-core:40641] [Bug #5539]

Revision 34108 - 12/23/2011 07:02 AM - kouji (Kouji Takao)

- ext/readline/readline.c (Init_readline): libedit check rl_getc_function only when rl_initialize() is called, and using_history() call rl_initialize(). This assignment should be placed before using_history(). [ruby-core:40641] [Bug #5539]

Revision 0239882c - 01/03/2012 04:54 AM - kosaki (Motohiro KOSAKI)

merge revision(s) 34108:

```
* ext/readline/readline.c (Init_readline): libedit check
rl_getc_function only when rl_initialize() is called, and
using_history() call rl_initialize(). This assignment should be
placed before using_history(). [ruby-core:40641] [Bug #5539]
```

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/branches/ruby_1_9_3@34196 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 34196 - 01/03/2012 04:54 AM - kosaki (Motohiro KOSAKI)

merge revision(s) 34108:

```
* ext/readline/readline.c (Init_readline): libedit check
rl_getc_function only when rl_initialize() is called, and
using_history() call rl_initialize(). This assignment should be
placed before using_history(). [ruby-core:40641] [Bug #5539]
```

History

#1 - 11/04/2011 01:30 PM - cjameshuff (Christopher Huff)

The issue pops up when Ruby uses the Editline library instead of GNU Readline. I suspect it is actually some conditionally compiled code that depends on the library being used, not the readline() call itself, since both the above workaround and GNU Readline work as expected.

#2 - 11/04/2011 02:18 PM - nagachika (Tomoyuki Chikanaga)

Hi,

In my environments, with GNU readline, your sample code work as expected.

I've found that a variable `rl_getc_function` is pointed `readline_getc()` and `readline_getc()` call `rb_funcall()` to invoke `IO#getbyte` method. Obviously the `IO#getbyte` release GVL during blocking.

I guess editline don't provide global variable `rl_getc_function` to customize `getc` function.

I think your workaround is potentially conflicted with `rl_getc_function` because it end up to call ruby method without GVL.

Thanks,

#3 - 11/04/2011 03:35 PM - nagachika (Tomoyuki Chikanaga)

Hi,

I'm sorry my previous estimate was wrong. nakada san told me that libeditline also has `rl_getc_function`. There should be another cause to prevent `readline_getc()` works as expected.

I'll examine it later.

Thanks,

#4 - 11/04/2011 11:46 PM - nagachika (Tomoyuki Chikanaga)

- File `readline.patch` added

- Category set to `ext`

- Assignee set to `kouji (Kouji Takao)`

I can reproduce the issue with `readline` configured with `--enable-libedit` option.

I wrote a patch.

Takao san, how about this patch?

Description of the patch:

libedit check `rl_getc_function` only when `rl_initialize()` is called. And `using_history()` may call `rl_initialize()`. So `rl_getc_function` should be assigned before calling `using_history()`.

Thanks,

#5 - 11/08/2011 12:24 AM - nagachika (Tomoyuki Chikanaga)

- File `readline_comment.patch` added

- Status changed from `Open` to `Assigned`

Obviously, the patch should contains comments about the constraint of the code sequence. Updated the patch.

#6 - 12/01/2011 11:55 PM - nagachika (Tomoyuki Chikanaga)

Takao san,

Is it hard to make time to take a look at this issue?

Thanks,

#7 - 12/23/2011 04:02 PM - kouji (Kouji Takao)

- Status changed from `Assigned` to `Closed`

- % Done changed from `0` to `100`

This issue was solved with changeset `r34108`.

Christopher, thank you for reporting this issue.

Your contribution to Ruby is greatly appreciated.

May Ruby be with you.

-
- `ext/readline/readline.c (Init_readline)`: libedit check `rl_getc_function` only when `rl_initialize()` is called, and `using_history()` call `rl_initialize()`. This assignment should be placed before `using_history()`. [`ruby-core:40641`] [Bug [#5539](#)]

#8 - 12/23/2011 05:32 PM - kouji (Kouji Takao)

- Tracker changed from `Bug` to `Backport`

- Project changed from `Ruby master` to `Backport193`

- Category deleted (`ext`)

- Status changed from `Closed` to `Open`

- Assignee deleted (`kouji (Kouji Takao)`)

Please backport r34108 to ruby_1_9_3

#9 - 01/03/2012 01:54 PM - kosaki (Motohiro KOSAKI)

- Status changed from Open to Closed

This issue was solved with changeset [r34196](#).
Christopher, thank you for reporting this issue.
Your contribution to Ruby is greatly appreciated.
May Ruby be with you.

merge revision(s) 34108:

```
* ext/readline/readline.c (Init_readline): libedit check
  rl_getc_function only when rl_initialize() is called, and
  using_history() call rl_initialize(). This assignment should be
  placed before using_history(). [ruby-core:40641] [Bug #5539]
```

Files

readline.patch	1009 Bytes	11/04/2011	nagachika (Tomoyuki Chikanaga)
readline_comment.patch	1.18 KB	11/08/2011	nagachika (Tomoyuki Chikanaga)