

Ruby trunk - Feature #6440

IO Marshal.load

05/16/2012 12:49 PM - Glass_saga (Masaki Matsushita)

Status:	Closed
Priority:	Normal
Assignee:	nobu (Nobuyoshi Nakada)
Target version:	2.0.0
Description	
<pre>=begin IO((Marshal.load))IO(IO)((IO#getbyte))IO(IO#read)) IO(IO)((IO))IO((Marshal.load))IO require 'benchmark' require 'tempfile' ary = Array.new(1000){ "hoge" } file = Tempfile.new("foo") Marshal.dump(ary, file) Benchmark.bm do x x.report do 100.times do file.rewind Marshal.load(file) end end end file.close rubytrunk trunk(r35660): user system total real 1.880000 0.000000 1.880000 (1.874681) proposed: user system total real 0.180000 0.000000 0.180000 (0.178556) patch =end</pre>	

Associated revisions

Revision 132eb25a - 11/20/2012 03:17 PM - glass

- marshal.c: add marshal readahead. marshaled Array, Hash and Struct have size at least number of its elements, marshal readahead will read the certain readable length and buffer when it needs more bytes. marshal readahead prevents many calls to IO#getbyte and IO#read, then it enables performance improvement. [ruby-dev:45637] [Feature #6440]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@37772 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 37772 - 11/20/2012 03:17 PM - glass

- marshal.c: add marshal readahead. marshaled Array, Hash and Struct have size at least number of its elements, marshal readahead will read the certain readable length and buffer when it needs more bytes. marshal readahead prevents many calls to IO#getbyte and IO#read, then it enables performance improvement. [ruby-dev:45637] [Feature #6440]

Revision 37772 - 11/20/2012 03:17 PM - Glass_saga (Masaki Matsushita)

- marshal.c: add marshal readahead. marshaled Array, Hash and Struct have size at least number of its elements, marshal readahead will read the certain readable length and buffer when it needs more bytes. marshal readahead prevents many calls to IO#getbyte and IO#read, then it enables performance improvement. [ruby-dev:45637] [Feature #6440]

Revision 37772 - 11/20/2012 03:17 PM - glass

- marshal.c: add marshal readahead. marshaled Array, Hash and Struct have size at least number of its elements, marshal readahead will read the certain readable length and buffer when it needs more bytes. marshal readahead prevents many calls to IO#getbyte and IO#read, then it enables performance improvement. [ruby-dev:45637] [Feature #6440]

Revision 37772 - 11/20/2012 03:17 PM - glass

- marshal.c: add marshal readahead. marshaled Array, Hash and Struct have size at least number of its elements, marshal readahead will read the certain readable length and buffer when it needs more bytes. marshal readahead prevents many calls to IO#getbyte and IO#read, then it enables performance improvement. [ruby-dev:45637] [Feature #6440]

Revision 37772 - 11/20/2012 03:17 PM - glass

- marshal.c: add marshal readahead. marshaled Array, Hash and Struct have size at least number of its elements, marshal readahead will read the certain readable length and buffer when it needs more bytes. marshal readahead prevents many calls to IO#getbyte and IO#read, then it enables performance improvement. [ruby-dev:45637] [Feature #6440]

Revision 37772 - 11/20/2012 03:17 PM - glass

- marshal.c: add marshal readahead. marshaled Array, Hash and Struct have size at least number of its elements, marshal readahead will read the certain readable length and buffer when it needs more bytes. marshal readahead prevents many calls to IO#getbyte and IO#read, then it enables performance improvement. [ruby-dev:45637] [Feature #6440]

History

#1 - 05/17/2012 12:57 AM - mame (Yusuke Endoh)

- Status changed from Open to Assigned
- Assignee set to nobu (Nobuyoshi Nakada)

XXXXXXXXXXXX

--
Yusuke Endoh mame@tsg.ne.jp

#2 - 05/17/2012 01:22 PM - nobu (Nobuyoshi Nakada)

```
=begin
XXXXXXXXXXXXXXXXXXXXXXXXXXXX
• (({struct load_arg.buf}))XXXXXXXXXX
• (({r_bytes1_partial}))&((readpartial))XXXXXXXXXX((read))XXXXXXXXXX((tmp_ptr))XXXX
```

XXXXXXXXXXXX

- (({arg->partial}))XXXXXXXXXXXXsymbolXXXXXXXXXXXX
- (({r_bytes}))&((r_bytes0()))XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
- XXXX(((if)))XXXXXXXXXXXXXXXXXXXX =end

#3 - 05/17/2012 01:27 PM - nobu (Nobuyoshi Nakada)

- Description updated

```
=begin
XXXXXX({s_getbyte})XXXXXXXXXXXX
=end
```

#4 - 05/18/2012 01:39 AM - Glass_saga (Masaki Matsushita)

XXXXXXXXXXXXXXXXXXXXXXXXXXXX

```
XXXMarshalXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
ungetcXXXXXXXXXXXXseekXXXXXXXXXXXXIOXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXIO#seekXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXIOXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

#5 - 05/18/2012 08:23 AM - ko1 (Koichi Sasada)

(2012/05/18 1:39), Glass_saga (Masaki Matsushita) wrote:

```
Marshal.ungetc seek IO#seek
```

Marshal

```
# Marshal.ungetc seek File
# Marshal.ungetc pipe
# Marshal.ungetc
```

```
Marshal.workaround Marshal
Marshal Marshal
```

```
require 'benchmark'
require 'tempfile'
```

```
ary = Array.new(10_000){ "hoge" }
file = Tempfile.new("foo")
file2 = Tempfile.new("bar")
Marshal.dump(ary, file)
Marshal.dump(Marshal.dump(ary), file2)
```

```
Benchmark.bm do |x|
  x.report do
    100.times do
      file.rewind
      Marshal.load(file)
    end
  end
  x.report do
    100.times do
      file2.rewind
      Marshal.load(Marshal.load(file2))
    end
  end
end
```

file.close

```
ruby 2.0.0dev (2012-05-04 trunk 35535) [i386-mswin32_100]
user system total real
6.068000 0.031000 6.099000 ( 6.334804)
0.530000 0.016000 0.546000 ( 0.573573)
```

Marshal

```
Marshal.load n byte
Marshal *1 m
n - m byte
```

*1: http://www.ruby-lang.org/ja/old-man/html/Marshal_A5D5A5A9A1BCA5DEA5C3A5C8.html

Marshal

--
// SASADA Koichi at atdot dot net

#6 - 06/09/2012 12:53 PM - akr (Akira Tanaka)

20120517 18:21 Tanaka Akira akr@fsij.org:

```
Marshal
marshal
```

0000000000000000

0000000000 tb 000000000000000000000000
0000000000

000000000000000000000000 marshal load 0000000000
000 process.c 00000000000000

rb_fork_err 0000000000 marshal 0000000000 errno 000000
0000000000000000000000000000000000

```
    if ((read(ep[0], &state, sizeof(state))) == sizeof(state) && state) {
        io = rb_io_fdopen(ep[0], O_RDONLY|O_BINARY, NULL);
        exc = rb_marshal_load(io);
        rb_set_errinfo(exc);
    }
```

```
#define READ_FROM_CHILD(ptr, len) \
(NIL_P(io) ? read(ep[0], (ptr), (len)) : rb_io_bufread(io, \
(ptr), (len)))
if ((size = READ_FROM_CHILD(&err, sizeof(err))) < 0) {
    err = errno;
}
if (size == sizeof(err) &&
errmsg && 0 < errmsg_bufllen) {
    ssize_t ret = READ_FROM_CHILD(errmsg, errmsg_bufllen-1);
    if (0 <= ret) {
        errmsg[ret] = '\0';
    }
}
```

00000000

--
[00 0][000 000][Tanaka Akira]

#7 - 06/11/2012 10:25 PM - Glass_saga (Masaki Matsushita)

- File patch2.diff added

00000000000000000000000000000000

00000000000000n000Array0m000000000000000000000000
000000n - m 000000000000000000

Array0000000000000000000000Array000000n - m 000000000000
00000000Array000000000000Array0000000000000100000000000000
n - m - 1 000000000000000000000000Array0000000000
0000000000Array00n - m 000000000000000000n - m - 1 000000000000

0000struct load_arg0readable00000000000000000000000000000000
r_object()0type0ARRAY00000000arg->readable0len - 10000100000000000000
0000n - m - 100000000000

00000000000000arg->readable + 100000000000
arg->readable + 100000000000Array000000n - m0000000000000000

Hash0Struct000000000000000000
0000000010000000002000000000arg->readable00(len - 1) * 200001000000200000000
00000Hash0Struct00000(n - m) * 2000000000arg->readable + 2000000000000
1000000000000r_byte0r_bytes0type0000000000000000000000000000

[ruby-dev:45637]00000000000000000000000000000000

trunk([r35983](#))
user system total real
0.560000 0.030000 0.590000 (0.601471)

proposed:
user system total real
0.090000 0.010000 0.100000 (0.113099)

patch00000000

#8 - 11/16/2012 03:49 PM - Glass_saga (Masaki Matsushita)

patch2.diff
patch2.diff

#9 - 11/21/2012 12:17 AM - Anonymous

- Status changed from Assigned to Closed
- % Done changed from 0 to 100

This issue was solved with changeset [r37772](#).
 Masaki, thank you for reporting this issue.
 Your contribution to Ruby is greatly appreciated.
 May Ruby be with you.

-
- marshal.c: add marshal readahead. marshaled Array, Hash and Struct have size at least number of its elements, marshal readahead will read the certain readable length and buffer when it needs more bytes. marshal readahead prevents many calls to IO#getbyte and IO#read, then it enables performance improvement. [ruby-dev:45637] [Feature [#6440](#)]

Files

patch.diff	4.61 KB	05/16/2012	Glass_saga (Masaki Matsushita)
patch2.diff	6.15 KB	06/11/2012	Glass_saga (Masaki Matsushita)