Ruby master - Bug #15469

Ruby2.6 included `bundler` does not handle specified `csv` gem.

12/27/2018 02:03 PM - watson1978 (Shizuo Fujita)

<table>
<thead>
<tr>
<th>Status:</th>
<th>Closed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority:</td>
<td>Normal</td>
</tr>
<tr>
<td>Assignee:</td>
<td>hsbt (Hiroshi SHIBATA)</td>
</tr>
<tr>
<td>Target version:</td>
<td></td>
</tr>
<tr>
<td>ruby -v:</td>
<td></td>
</tr>
<tr>
<td>Backport:</td>
<td>2.4: DONTNEED, 2.5: DONTNEED, 2.6: DONE</td>
</tr>
</tbody>
</table>

Description
I want to use csv library which is same version of Ruby 2.5.
Because Ruby 2.6's csv library has some breaking changes and they break our product.
However, Ruby 2.6's bundler does not handle csv which I specified the version.

Test code

- Gemfile

```ruby
source "https://rubygems.org"

gem 'roo'
gem 'csv', '1.0.0'
```

- test.rb

```ruby
require "bundler/setup"
require "csv"
p CSV::VERSION
```

Result

```ruby
$ bundle exec ruby -v test.rb
ruby 2.6.0p0 (2018-12-25 revision 66547) [x86_64-darwin18]
"3.0.2"
```

CSV::VERSION shows 3.0.2 and it indicates that Ruby 2.6 included csv library was used.

Workarounds

If I install the bundler gem through gem install bundler, it works well.

```ruby
$ gem install bundler
Fetching bundler-1.17.3.gem
Successfully installed bundler-1.17.3
1 gem installed
```

```ruby
$ bundle exec ruby -v test.rb
ruby 2.6.0p0 (2018-12-25 revision 66547) [x86_64-darwin18]
"2.4.8"
```

Related issues:
Related to Ruby master - Bug #15500: Behavior of require method in 2.5 is dif...
Closed

Associated revisions

Revision 0fad484 - 02/05/2019 04:58 AM - hsbt (Hiroshi SHIBATA)
Ignore to add bundler lib direcotry if it is same as rubylibdir.
Revision 67006 - 02/05/2019 04:58 AM - hsbt (Hiroshi SHIBATA)
Ignore to add bundler lib directory if it is same as rubylibdir.

Revision e0d63927 - 03/06/2019 07:47 AM - naruse (Yui NARUSE)
merge revision(s) 67006: [Backport #15469]
Ignore to add bundler lib directory if it is same as rubylibdir.

Revision 67179 - 03/06/2019 07:47 AM - naruse (Yui NARUSE)
merge revision(s) 67006: [Backport #15469]
Ignore to add bundler lib directory if it is same as rubylibdir.

What's the version of csv library that is you expected?
I wonder csv-1.0.0 on your Gemfile.

I need that CSV::VERSION indicates 2.4.8 as well as Workaround.

I could reproduce too with "Test code".
Also, I could reproduce with simpler code below.

Gemfile

```
source "https://rubygems.org"
gem "csv", "1.0.0"
```

test.rb

```
require "csv"
p csv: CSV::VERSION
```

results

```
$ ruby2.6 -v && bundle2.6 -v
ruby 2.6.0p0 (2018-12-25 revision 66547) [x86_64-linux]
Bundler version 1.17.2
$ bundle2.6 exec ruby2.6 test.rb
 [:csv=>"3.0.2"] # expects "2.4.8" for 1.0.0 gem https://github.com/ruby/csv/blob/v1.0.0/lib/csv.rb#L211
```

In addition, it does not seem to be limited to "csv" gem.
I confirmed that the same problem occurs with "json".

#4 - 01/23/2019 02:46 PM - aeroastro (Takumasa Ochi)
I have found out the root cause of this.

Root Cause

- Bundler has become a built-in gem since Ruby 2.6.
- $LOAD_PATH required for bundler is moved from custom gem path (e.g. lib/ruby/gems/2.5.0/gems/bundler-1.17.2/lib ) to ruby built-in gem path (lib/ruby/2.6.0)
- In the same path, csv, json, and other built-in gems are located.
- When invoked via bundler, this path for bundler is inserted at the head of $LOAD_PATH, which takes precedence over all the other paths.
- These built-in gems are loaded instead of those specified by Gemfile.

Why Workaround Works

When bundler is updated by gem install, the $LOAD_PATH is moved from ruby built-in gem path to custom gem path. This fixes shared built-in gem path issue.

Suggested Fix

IMHO, moving bundler to isolated path instead of shared built-in path would fix this issue.

Bug Detail

Following are the code to reproduce and explain this issue.

Gemfile

```ruby
source "https://rubygems.org"

gem 'json', '-> 1.8.6'
```

Code

```ruby
before = $LOADED_FEATURES.dup
require 'json'
after = $LOADED_FEATURES.dup
require 'yaml'

puts YAML.dump(
  ruby: RUBY_VERSION,
  bundler: (Bundler::VERSION rescue nil),
  json: JSON::VERSION,
  load_path: $LOAD_PATH,
  added_features: after - before,
)

JSON.dump("GET")
```

Result

```
# I have installed ruby with rbenv and ruby-build
# 'bundle install --path=vendor/bundle' is executed as prerequisites

$ rbenv shell 2.5.3 && bundle exec ruby test.rb
---
:ruby: 2.5.3
:bundler: 1.17.2
:json: 1.8.6
:load_path:
  - "/Users/user-name/.rbenv/versions/2.5.3/lib/ruby/gems/2.5.0/gems/bundler-1.17.2/lib"
  - "/Users/user-name/.rbenv/dexec/gem-rehash"
  - "/Users/user-name/Temp/alienvendor/bundle/ruby/2.5.0/gems/json-1.8.6/lib"
  - "/Users/user-name/Temp/alienvendor/bundle/ruby/2.5.0/extensions/x86_64-darwin-16/2.5.0-static/json-1.8.6"
  - "/Users/user-name/.rbenv/versions/2.5.3/lib/ruby/gems/2.5.0/gems/bundler-1.17.2/lib/gems/bundler-1.17.2/lib"
  - "/Users/user-name/.rbenv/versions/2.5.3/lib/ruby/site_ruby/2.5.0"
  - "/Users/user-name/.rbenv/versions/2.5.3/lib/ruby/site_ruby/2.5.0/x86_64-darwin16"
  - "/Users/user-name/.rbenv/versions/2.5.3/lib/ruby/site_ruby"
  - "/Users/user-name/.rbenv/versions/2.5.3/lib/ruby/vendor_ruby/2.5.0"
  - "/Users/user-name/.rbenv/versions/2.5.3/lib/ruby/vendor_ruby/2.5.0/x86_64-darwin16"
  - "/Users/user-name/.rbenv/versions/2.5.3/lib/ruby/vendor_ruby"
$ rbenv shell 2.6.0 && bundle exec ruby test.rb

---
ruby: 2.6.0
bundler: 1.17.3
json: 1.8.6
load_path:
  - "/Users/user-name/.rbenv/versions/2.6.0/lib/ruby/gems/2.6.0/gems/bundler-1.17.3/lib"
  - "Users/user-name/.rbenv/rbenv.d/exec/gem-rehash"
  - "Users/user-name/temp/aiueo/vendor/bundle/ruby/2.6.0/gems/json-1.8.6/lib"
  - "Users/user-name/temp/aiueo/vendor/bundle/extensions/x86_64-darwin-16/2.6.0-static/json-1.8.6"
  - "Users/user-name/.rbenv/versions/2.6.0/lib/ruby/gems/bundler-1.17.3/lib/gems/bundler-1.17.3/lib"
  - "Users/user-name/.rbenv/versions/2.6.0/lib/ruby/site_ruby/2.6.0"
  - "Users/user-name/.rbenv/versions/2.6.0/lib/ruby/site_ruby/2.6.0/x86_64-darwin64"
  - "Users/user-name/.rbenv/versions/2.6.0/lib/ruby/vendordir/ruby/2.6.0"
  - "Users/user-name/.rbenv/versions/2.6.0/lib/ruby/vendordir/ruby/2.6.0/x86_64-darwin64"
  - "Users/user-name/.rbenv/versions/2.6.0/lib/ruby/2.6.0/json/version.rb"
  - "Users/user-name/.rbenv/versions/2.6.0/lib/ruby/2.6.0/ostruct.rb"
  - "Users/user-name/.rbenv/versions/2.6.0/lib/ruby/2.6.0/json/generic_object.rb"
  - "Users/user-name/.rbenv/versions/2.6.0/lib/ruby/2.6.0/json/common.rb"
  - "Users/user-name/.rbenv/versions/2.6.0/lib/ruby/2.6.0/x86_64-darwin64/enc/utf_16be.bundle"
  - "Users/user-name/.rbenv/versions/2.6.0/lib/ruby/2.6.0/x86_64-darwin64/enc/utf_16le.bundle"
  - "Users/user-name/.rbenv/versions/2.6.0/lib/ruby/2.6.0/x86_64-darwin64/enc/utf_32be.bundle"
  - "Users/user-name/.rbenv/versions/2.6.0/lib/ruby/2.6.0/x86_64-darwin64/enc/utf_32le.bundle"
  - "Users/user-name/.rbenv/versions/2.6.0/lib/ruby/2.6.0/json/ext/parser.bundle"
  - "Users/user-name/.rbenv/versions/2.6.0/lib/ruby/2.6.0/json/ext/generator.bundle"
  - "Users/user-name/.rbenv/versions/2.6.0/lib/ruby/2.6.0/json/ext.rb"
  - "Users/user-name/.rbenv/versions/2.6.0/lib/ruby/2.6.0/json.rb"

Traceback (most recent call last):
  3: from test.rb:15:in `main'
  2: from /Users/user-name/.rbenv/versions/2.6.0/lib/ruby/2.6.0/json/common.rb:394:in `dump'
  1: from /Users/user-name/.rbenv/versions/2.6.0/lib/ruby/2.6.0/json/common.rb:224:in `generate'
/Users/user-name/.rbenv/versions/2.6.0/lib/ruby/2.6.0/json/common.rb:224:in `generate': only generation of JSO
N objects or arrays allowed (JSON::GeneratorError)

$ gem install bundler -v 1.17.3
Fetching bundler-1.17.3.gem
Successfully installed bundler-1.17.3
Parsing documentation for bundler-1.17.3
Installing ri documentation for bundler-1.17.3
Done installing documentation for bundler after 3 seconds
1 gem installed

$ rbenv shell 2.6.0 && bundle exec ruby test.rb
---
ruby: 2.6.0
bundler: 1.17.3
json: 1.8.6
load_path:
  - "Users/user-name/.rbenv/versions/2.6.0/lib/ruby/gems/2.6.0/gems/bundler-1.17.3/lib"
  - "Users/user-name/.rbenv/versions/2.6.0/lib/ruby/gems/bundler-1.17.3/lib/gems/bundler-1.17.3/lib"
  - "Users/user-name/.rbenv/versions/2.6.0/lib/ruby/site_ruby/2.6.0"
  - "Users/user-name/.rbenv/versions/2.6.0/lib/ruby/site_ruby/2.6.0/x86_64-darwin64"
  - "Users/user-name/.rbenv/versions/2.6.0/lib/ruby/2.6.0/json/version.rb"
  - "Users/user-name/.rbenv/versions/2.6.0/lib/ruby/2.6.0/ostruct.rb"
  - "Users/user-name/.rbenv/versions/2.6.0/lib/ruby/2.6.0/json/generic_object.rb"
  - "Users/user-name/.rbenv/versions/2.6.0/lib/ruby/2.6.0/json/common.rb"
  - "Users/user-name/.rbenv/versions/2.6.0/lib/ruby/2.6.0/x86_64-darwin64/enc/utf_16be.bundle"
  - "Users/user-name/.rbenv/versions/2.6.0/lib/ruby/2.6.0/x86_64-darwin64/enc/utf_16le.bundle"
  - "Users/user-name/.rbenv/versions/2.6.0/lib/ruby/2.6.0/x86_64-darwin64/enc/utf_32be.bundle"
  - "Users/user-name/.rbenv/versions/2.6.0/lib/ruby/2.6.0/x86_64-darwin64/enc/utf_32le.bundle"
  - "Users/user-name/.rbenv/versions/2.6.0/lib/ruby/2.6.0/json/ext/parser.bundle"
  - "Users/user-name/.rbenv/versions/2.6.0/lib/ruby/2.6.0/json/ext/generator.bundle"
  - "Users/user-name/.rbenv/versions/2.6.0/lib/ruby/2.6.0/json/ext.rb"
  - "Users/user-name/.rbenv/versions/2.6.0/lib/ruby/2.6.0/json.rb"
You can see the first element in $LOAD_PATH is problematic when built-in bundler is used. (/Users/user-name/.rbenv/versions/2.6.0/lib/ruby/2.6.0)

(json in shared built-in gem path is required instead of Gemfile-specified json gem.

(The exception above is caused by mixed versions of json gem.)

#5 - 01/23/2019 10:31 PM - Eregon (Benoit Daloze)
Should Bundler be a bundled gem (and not a default gem)?)
I think this would help a lot, and allow uninstalling the preinstalled bundler if needed.

As mentioned above, it would also keep the Bundler implementation simple by being able to assume it’s in a gem-like layout.

#6 - 01/31/2019 12:28 AM - hsbt (Hiroshi SHIBATA)
- Related to Bug #15500: Behavior of require method in 2.5 is different from 2.4 and 2.6 added

#7 - 01/31/2019 06:13 AM - hsbt (Hiroshi SHIBATA)
@watson1978 (Shizuo Fujita)
Can you try this with Ruby 2.6.1? r66867 fixed this issue maybe.

#8 - 01/31/2019 09:45 AM - watson1978 (Shizuo Fujita)
hsbt (Hiroshi SHIBATA) wrote:

Can you try this with Ruby 2.6.1? r66867 fixed this issue maybe.

Thank you for your working.
I re-tried the test code in description,
However, there is not difference for me....

$ bundle install
Fetching gem metadata from https://rubygems.org/........
Resolving dependencies...
Using bundler 1.17.2
Fetching rubyzip 1.2.2
Fetching csv 1.0.0
Fetching mini_portile2 2.4.0
Installing csv 1.0.0
Installing mini_portile2 2.4.0
Installing rubyzip 1.2.2
Fetching nokogiri 1.10.1
Installing nokogiri 1.10.1 with native extensions
Fetching roo 2.8.1
Installing roo 2.8.1
Bundle complete! 2 Gemfile dependencies, 6 gems now installed.
Use `bundle info [gemname]` to see where a bundled gem is installed.

$ bundle exec ruby --v test.rb
ruby 2.6.1p33 (2019-01-30 revision 66950) [x86_64-darwin18]
"3.0.4"

09/01/2022
#9 - 02/02/2019 08:31 AM - hsbt (Hiroshi SHIBATA)
- File 15469-bundler-loadpath.patch added

Thanks all.

I investigate this issue with [https://bugs.ruby-lang.org/issues/15469#note-4](https://bugs.ruby-lang.org/issues/15469#note-4) and fixed this issue on Bundler::SharedHelper.set_rubylib.

Does anyone try with the attached patch?

#10 - 02/02/2019 09:54 AM - watson1978 (Shizuo Fujita)

Does anyone try with the attached patch?

I've got an expected result finally with the patch.
Thank you for great works!!!

$ bundle exec ruby -v test.rb
ruby 2.6.1p33 (2019-01-30 revision 66950) [x86_64-darwin18] "2.4.8"

$ ruby -v test.rb
ruby 2.6.1p33 (2019-01-30 revision 66950) [x86_64-darwin18] "2.4.8"

#11 - 02/04/2019 05:49 AM - hsbt (Hiroshi SHIBATA)

- Backport changed from 2.4: UNKNOWN, 2.5: UNKNOWN, 2.6: UNKNOWN to 2.4: DONTNEED, 2.5: DONTNEED, 2.6: REQUIRED

#12 - 02/04/2019 03:38 PM - aeroastro (Takumasa Ochi)

Does anyone try with the attached patch?

I've also got the ideal result as follows. (with patched ruby 2.6.1)
Now there is no issue on $LOAD_PATH.

Thank you very much for your great work!

$ rbenv shell 2.6.1 && ruby --version && bundle exec ruby test.rb
ruby 2.6.1p33 (2019-01-30 revision 66950) [x86_64-darwin16]
---
ruby: 2.6.1
bundler: 1.17.2
json: 1.8.6
load_path:
  - "/Users/user-name/.rbenv/rbenv.d/exec/gem-rehash"
  - "/Users/user-name/temp/aiueo/vendor/bundle/ruby/2.6.0/gems/json-1.8.6/lib"
  - "/Users/user-name/temp/aiueo/vendor/bundle/ruby/2.6.0/extensions/x86_64-darwin-16/2.6.0-static/json-1.8.6"
  - "/Users/user-name/.rbenv/versions/2.6.1/lib/ruby/gems/2.6.0/gems/bundler-1.17.2/lib"
  - "/Users/user-name/.rbenv/versions/2.6.1/lib/ruby/site_ruby/2.6.0"
  - "/Users/user-name/.rbenv/versions/2.6.1/lib/ruby/site_ruby/2.6.0/x86_64-darwin16"
  - "/Users/user-name/.rbenv/versions/2.6.1/lib/ruby/vendor_ruby/2.6.0/x86_64-darwin16"
  - "/Users/user-name/.rbenv/versions/2.6.1/lib/ruby/vendor_ruby/2.6.0"
  - "/Users/user-name/.rbenv/versions/2.6.1/lib/ruby/site_ruby/2.6.0"
  - "/Users/user-name/.rbenv/versions/2.6.1/lib/ruby/vendor_ruby/2.6.0/x86_64-darwin16"
  - "/Users/user-name/.rbenv/versions/2.6.1/lib/ruby/vendor_ruby/2.6.0"
  - "/Users/user-name/.rbenv/versions/2.6.1/lib/ruby/vendor_ruby/2.6.0/x86_64-darwin16"
  - "/Users/user-name/.rbenv/versions/2.6.1/lib/ruby/2.6.0/x86_64-darwin16"
:added_features:
  - "/Users/user-name/temp/aiueo/vendor/bundle/ruby/2.6.0/gems/json-1.8.6/lib/json/version.rb"
  - "/Users/user-name/.rbenv/versions/2.6.1/lib/ruby/2.6.0/ostruct.rb"
  - "/Users/user-name/temp/aiueo/vendor/bundle/ruby/2.6.0/gems/json-1.8.6/lib/gems/json.json-1.8.6.rb"
  - "/Users/user-name/temp/aiueo/vendor/bundle/ruby/2.6.0/gems/json-1.8.6/lib/gems/bundler-1.17.2/lib"
  - "/Users/user-name/.rbenv/versions/2.6.1/lib/ruby/site_ruby/2.6.0/x86_64-darwin16/enc/utf_16be.bundle"
  - "/Users/user-name/.rbenv/versions/2.6.1/lib/ruby/site_ruby/2.6.0/x86_64-darwin16/enc/utf_32be.bundle"
  - "/Users/user-name/.rbenv/versions/2.6.1/lib/ruby/site_ruby/2.6.0/x86_64-darwin16/enc/utf_16le.bundle"
  - "/Users/user-name/.rbenv/versions/2.6.1/lib/ruby/site_ruby/2.6.0/x86_64-darwin16/enc/utf_32le.bundle"
#13 - 02/05/2019 04:58 AM - hsbt (Hiroshi SHIBATA)
- Status changed from Assigned to Closed


Ignore to add bundler lib directory if it is same as rubylibdir.

[Bug #15469](ruby-core:90742)

#14 - 02/05/2019 06:22 AM - hsbt (Hiroshi SHIBATA)
- File r15469-bundler-final.patch added

I attached the patch of the upstream fixes.

The bundler team will ship Bundler 1.7.4/2.0.2 with this patch.

#15 - 03/06/2019 07:47 AM - naruse (Yui NARUSE)
- Backport changed from 2.4: DONTNEED, 2.5: DONTNEED, 2.6: REQUIRED to 2.4: DONTNEED, 2.5: DONTNEED, 2.6: DONE

ruby_2_6 r67179 merged revision(s) 67006.

**Files**

<table>
<thead>
<tr>
<th>Patch Name</th>
<th>Size</th>
<th>Date</th>
<th>Author</th>
</tr>
</thead>
<tbody>
<tr>
<td>15469-bundler-loadpath.patch</td>
<td>528 Bytes</td>
<td>02/02/2019</td>
<td>hsbt</td>
</tr>
<tr>
<td>r15469-bundler-final.patch</td>
<td>1.69 KB</td>
<td>02/05/2019</td>
<td>hsbt</td>
</tr>
</tbody>
</table>