Ruby trunk - Bug #5060

Executables in bin folder conflict with their gem versions.

07/20/2011 09:29 PM - vo.x (Vit Ondruch)

Status:	Closed		
Priority:	Normal		
Assignee:	hsbt (Hiroshi SHIBATA)		
Target version:			
ruby -v:	1.9.3	**Backport:**	

Description
It would be nice if the executables for rdoc, rake, ri, etc. locate in bin folder, would be standard RubyGems stubs. Otherwise, once there is installed version of such library as a gem, the original file is rewritten and the functionality of bundled library is lost.
But my main concern is the conflict between ruby and gem executables, since it makes packaging for Linux distributions hard (for Fedora and RHEL in my case).

Related issues:			
Related to Ruby trunk - Bug #4485: Gem.bin_path('rake') does not return the p...		**Closed**	**03/09/2011**

Associated revisions

Revision a9433281 - 11/25/2018 01:20 AM - hsbt (Hiroshi SHIBATA)

Use stub executables generated by RubyGems istead of original executables.
It resolved the conflict issues when invoking gem i rdoc and the binstub
issues with Bundler and Rails.

[Bug #5060][ruby-core:38257][Fix GH-2023]

- https://github.com/rubygems/rubygems/pull/2338
- https://github.com/heroku/heroku-buildpack-ruby/issues/829

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@65963 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 65963 - 11/25/2018 01:20 AM - hsbt (Hiroshi SHIBATA)

Use stub executables generated by RubyGems istead of original executables.
It resolved the conflict issues when invoking gem i rdoc and the binstub
issues with Bundler and Rails.

[Bug #5060][ruby-core:38257][Fix GH-2023]

- https://github.com/rubygems/rubygems/pull/2338
- https://github.com/heroku/heroku-buildpack-ruby/issues/829

Revision 65963 - 11/25/2018 01:20 AM - hsbt (Hiroshi SHIBATA)

Use stub executables generated by RubyGems istead of original executables.
It resolved the conflict issues when invoking gem i rdoc and the binstub
issues with Bundler and Rails.

[Bug #5060][ruby-core:38257][Fix GH-2023]

- https://github.com/rubygems/rubygems/pull/2338
- https://github.com/heroku/heroku-buildpack-ruby/issues/829

History

#1 - 07/25/2011 08:06 PM - naruse (Yui NARUSE)

- Status changed from Open to Assigned

- Assignee set to drbrain (Eric Hodel)

- Target version set to 2.0.0

#2 - 07/26/2011 08:50 AM - drbrain (Eric Hodel)

- Status changed from Assigned to Closed

Since I committed [r32608](#) and r32611 I think this issue is now invalid.

If a different version of rake or rdoc are installed they will behave the same as the bundled library. rake 0.9.2.1 will load the built-in version of rake, for example. If a newer version of RDoc were installed rdoc and ri would behave the same.

I'm closing this ticket unless differing behavior can be reproduced.

#3 - 06/21/2012 02:34 PM - vo.x (Vit Ondruch)

- Status changed from Closed to Assigned

Sorry, I'm reopening, but the issue is still present.

When you install Ruby, it carries initial version of /usr/bin/rdoc 1. Now let's assume that you will install updated RDoc, which replaces the file with the typical RubyGems stub, something like:

```
$ cat /usr/bin/rdoc
#!/usr/bin/ruby
#
# This file was generated by RubyGems.
#
# The application 'rdoc' is installed as part of a gem, and
# this file is here to facilitate running it.
#

require 'rubygems'

version = ">= 0"

if ARGV.first
  str = ARGV.first
  str = str.dup.force_encoding("BINARY") if str.respond_to? :force_encoding
  if str =~ /\A_(.*)_\z/
    version = $1
    ARGV.shift
  end
end

gem 'rdoc', version
load Gem.bin_path('rdoc', 'rdoc', version)
```

Now, this is problem for packaging systems, namely RPM (but I would be surprised if that is not issue for DEB as well). If you will do monolithic Ruby package, which provides /usr/bin/rdoc and later you want to package independent updated version of RDoc gem, which provides /usr/bin/rdoc as well, you are in conflict, since the files are different. If they would be the same (and there is no reason to be different IMO), there would be no conflict. The file would be own by two packages in parallel, which is supported scenario.

So what I'd like to see is the RubyGems stub in /usr/bin, especially after you committed [r32608](#).

Actually, I'm trying to put together some proposal to solve [#5481](#) and [#6124](#), which might fix also this issue.

Thank you.

[1] https://github.com/ruby/ruby/blob/trunk/bin/rdoc

#4 - 02/17/2013 05:35 PM - drbrain (Eric Hodel)

- Target version changed from 2.0.0 to 2.6

This is too risky to fix for 2.0.0, sorry I did not have time.

#5 - 10/14/2013 10:20 PM - salimane (Salimane Adjao Moustapha)

drbrain (Eric Hodel) wrote:

> This is too risky to fix for 2.0.0, sorry I did not have time.
> I would like to know when this issue can be solved and pushed into trunk at least :)
> Thanks

#6 - 12/08/2017 02:12 PM - mame (Yusuke Endoh)

- Assignee changed from drbrain (Eric Hodel) to hsbt (Hiroshi SHIBATA)

@hsbt-san, I might be wrong but has this issue been solved by default/bundled gems?

#7 - 12/12/2017 02:54 PM - vo.x (Vit Ondruch)

No, this is still not resolved (testing with r61169). Only Rake (which is "bundled" gem) appears to have RubyGems generated stub which won't conflict unless RubyGems change the algorithm. The remaining executables (namely bundle, bundler, rdoc) are conflicting.

#8 - 12/13/2017 01:55 AM - hsbt (Hiroshi SHIBATA)

I understood this issue. I try to resolve current specification at Ruby 2.6.

#9 - 12/25/2017 06:15 PM - naruse (Yui NARUSE)

- Target version deleted (2.6)

#10 - 11/23/2018 02:13 AM - hsbt (Hiroshi SHIBATA)

I created a patch for fixing this.

https://github.com/ruby/ruby/pull/2023

nobu (Nobuyoshi Nakada) Can you review the above patch?

#11 - 11/25/2018 01:20 AM - hsbt (Hiroshi SHIBATA)

- Status changed from Assigned to Closed

Applied in changeset trunk|r65963.

Use stub executables generated by RubyGems istead of original executables.
It resolved the conflict issues when invoking gem i rdoc and the binstub
issues with Bundler and Rails.

[Bug #5060][ruby-core:38257][Fix GH-2023]

- https://github.com/rubygems/rubygems/pull/2338
- https://github.com/heroku/heroku-buildpack-ruby/issues/829