GMOペパボ

How to develop the Standard Libraries of Ruby?

The future of Ruby ecosystem with GitHub

Hiroshi SHIBATA / GMO Pepabo, Inc.

2021.09.11 RubyKaigi takeout 2021

GMOペパボ



Hiroshi SHIBATA @hsbt

Executive Officer VP of Engineering
Technical Director
at GMO Pepabo, Inc. @pepabo

https://www.hsbt.org

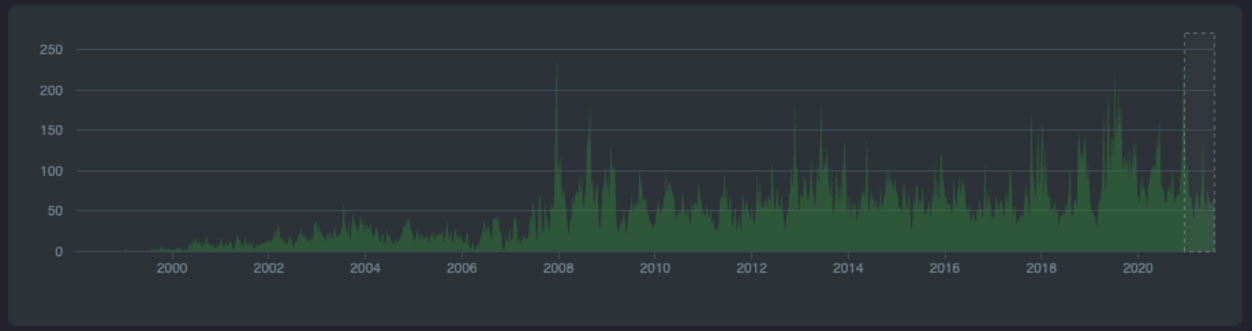
Pulse
Contributors
Community
Traffic
Commits
Code frequency
Dependency graph
Network
Forks

People

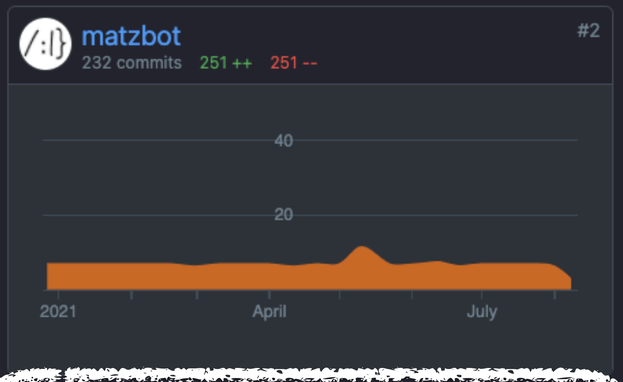
Dec 25, 2020 – Aug 11, 2021

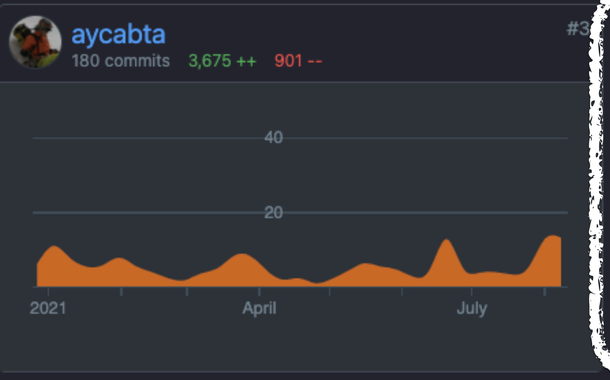
Contributions: Commits

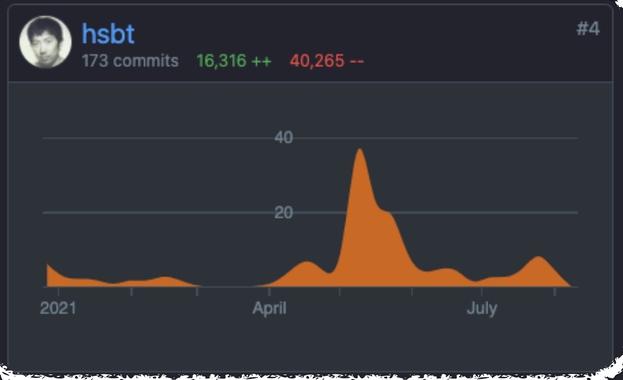
Contributions to master, excluding merge commits and bot accounts











Agenda

- The introduction for Gemification of Ruby 3.0, RubyGems 3.2 and Bundler 2.2
- The details of developing the default gems
- The future planf for Ruby 3.1/4.0



The introduction for Gemification of Ruby 3.0, RubyGems 3.2 and Bundler 2.2

Gemification of Ruby 3.0

This matrix shows number of standard libraries and their classifications in Ruby 3.0.

	Standard Libraries	Default Gems	Bundled Gems
Pure Ruby	3	52	8
C extensions	6	20	0

With ruby/ruby repository

Gemification on Ruby 3.1(TBD)

I will extract the following default gems to bundled gems.

```
net-ftp                 net-pop                 matrix
net-imap                net-smtp                prime
Debug
```

And I gave up to promote the following default gems.

```
English - Rubocop still suggest to use it.
ostruct - A lot of gems depends on it.
date    - Rails dependes on it.
…
```



Which stdlibs will remain in the stdlibs?

I decided to not promote the following gems to the default gems.

```
mkmf            coverage        ripper
rbconfig        monitor         socket
objspace
```



Native extension with the bundled gems

- We couldn’t promote the native extension to the bundled gems now.
- We need to use installed Ruby before its installed.

```
977
978  install?(:ext, :comm, :gem, :'bundled-gems') do
979    gem_dir = Gem.default_dir
980    install_dir = with_destdir(gem_dir)
981    prepare "bundled gems", gem_dir
982    RbInstall.no_write do
983      makedirs(Gem.ensure_gem_subdirectories(install_dir, $dir_mode).map {|d| File.join(gem_dir, d)})
984    end
985
986    installed_gems = {}
987    options = {
988      :install_dir => install_dir,
989      :bin_dir => with_destdir(bindir),
990      :domain => :local,
991      :ignore_dependencies => true,
992      :dir_mode => $dir_mode,
993      :data_mode => $data_mode,
994      :prog_mode => $script_mode,
995      :wrappers => true,
996      :format_executable => true,
997    }
998    gem_ext_dir = "#$extout/gems/#{CONFIG['arch']}"
999    extensions_dir = Gem::StubSpecification.gemspec_stub("", gem_dir, gem_dir).extensions_dir
1000   File.foreach("#{srcdir}/gems/bundled_gems") do |name|
1001     next if /^\s*(?:#|$)/ =~ name
1002     next unless /^(\S+)\s+(\S+).*/ =~ name
1003     gem_name = "#$1-#$2"
1004     path = "#{srcdir}/.bundle/gems/#{gem_name}/#{gem_name}.gemspec"
1005     if File.exist?(path)
1006       spec = load_gemspec(path)
1007     else
1008       path = "#{srcdir}/.bundle/gems/#{gem_name}/#$1.gemspec"
1009       next unless File.exist?(path)
1010       spec = load_gemspec(path, true)
1011     end
1012     next unless spec.platform == Gem::Platform::RUBY
1013     next unless spec.full_name == gem_name
1014     spec.extension_dir = "#{extensions_dir}/#{spec.full_name}"
1015     if File.directory?(ext = "#{gem_ext_dir}/#{spec.full_name}")
1016       spec.extensions[0] ||= "-"
1017     end
1018     package = RbInstall::DirPackage.new spec
1019     ins = RbInstall::UnpackedInstaller.new(package, options)
```

The new feature of RG 3.2/Bundler 2.2

- Fixed the priority for multi-source dependencies for CVE-2020-36327 at “> Bundler 2.2.17”.
- Support Multi-platform.
  - You should specify it with `bundle lock --platform …`
- [ANN] I got the privilege for handling the rubygems.org



The details of the default gems

How develop the default gems

- Default gems repository is located under the https://github.com/ruby
- It has a GitHub actions, enabled pull-request. So, you can develop typically github workflow.



The differences with ruby/ruby and ruby/*

You can run the test suite with under the repository.

ruby/ruby

```
$ autoreconf
$ ./configure —disable-install-doc
$ make -j
$ make check
```

ruby/*

```
$ bundle install
$ rake test
```



How merge the default gems to ruby/ruby

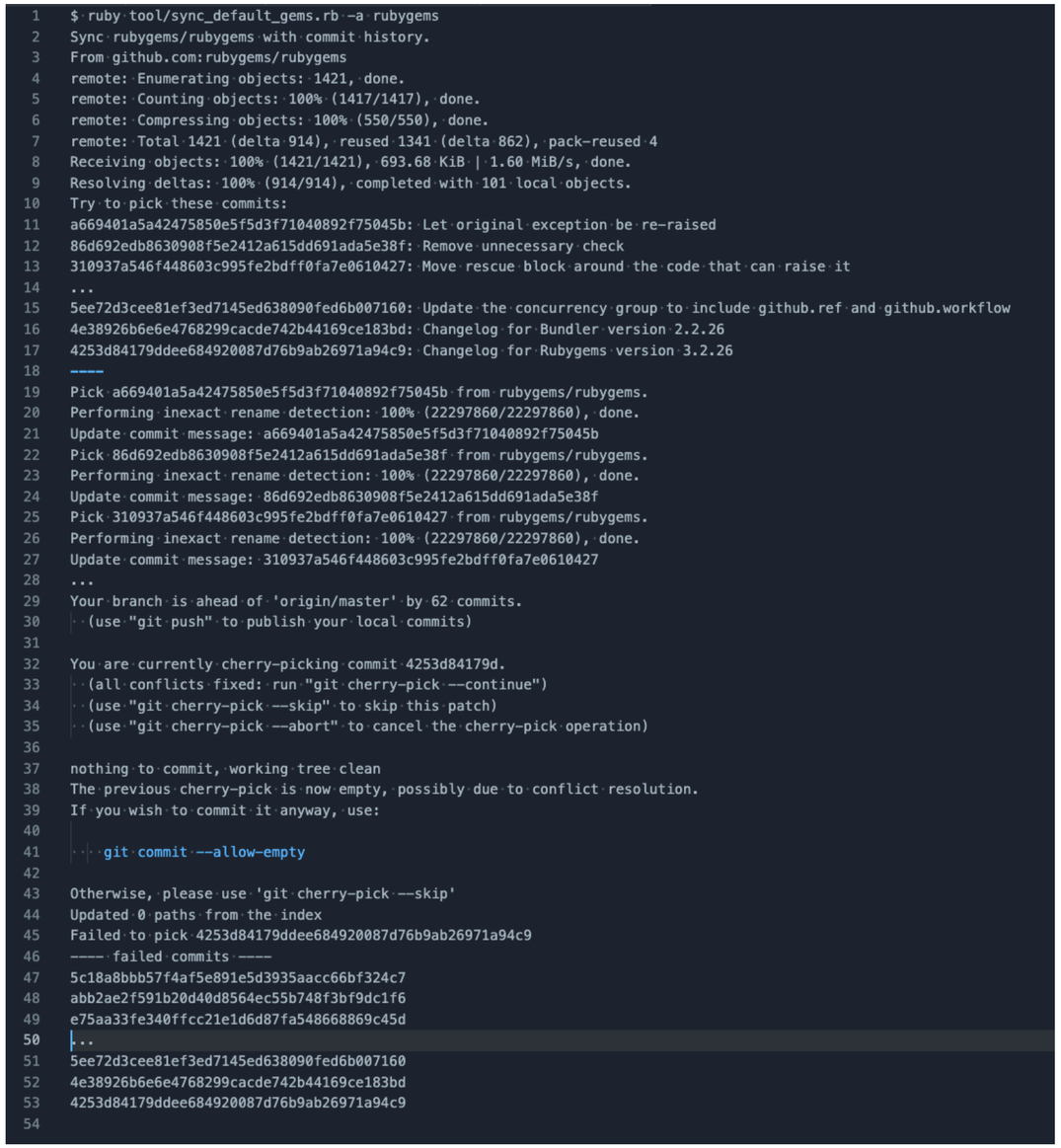
- Default gems side(ruby/*)
  - Copy ruby/ruby. Run `make check`, If tests failed, I fixed them and backport to default gems.
- Stdlibs side(ruby/ruby)
  - Backport ruby/*. Run `rake test`, If tests failed, I fixed them and forwardport to stdlibs.



sync_default_gems.rb

This script manage the commits between ruby/ruby and ruby/*.

- Pick single commit or commits from ruby/* to ruby/ruby.
- Add URL for the original commits of ruby/* to commit message.



```
$ ruby tool/sync_default_gems.rb -a rubygems
Sync rubygems/rubygems with commit history.
From github.com:rubygems/rubygems
remote: Enumerating objects: 1421, done.
remote: Counting objects: 100% (1417/1417), done.
remote: Compressing objects: 100% (550/550), done.
remote: Total 1421 (delta 914), reused 1341 (delta 862), pack-reused 4
Receiving objects: 100% (1421/1421), 693.68 KiB | 1.60 MiB/s, done.
Resolving deltas: 100% (914/914), completed with 101 local objects.
Try to pick these commits:
a669401a5a42475850e5f5d3f71040892f75045b: Let original exception be re-raised
86d692edb8630908f5e2412a615dd691ada5e38f: Remove unnecessary check
310937a546f448603c995fe2bdff0fa7e0610427: Move rescue block around the code that can raise it
...
5ee72d3cee81ef3ed7145ed638090fed6b007160: Update the concurrency group to include github.ref and github.workflow
4e38926b6e6e4768299cacde742b44169ce183bd: Changelog for Bundler version 2.2.26
4253d84179ddee684920087d76b9ab26971a94c9: Changelog for Rubygems version 3.2.26
----
Pick a669401a5a42475850e5f5d3f71040892f75045b from rubygems/rubygems.
Performing inexact rename detection: 100% (22297860/22297860), done.
Update commit message: a669401a5a42475850e5f5d3f71040892f75045b
Pick 86d692edb8630908f5e2412a615dd691ada5e38f from rubygems/rubygems.
Performing inexact rename detection: 100% (22297860/22297860), done.
Update commit message: 86d692edb8630908f5e2412a615dd691ada5e38f
Pick 310937a546f448603c995fe2bdff0fa7e0610427 from rubygems/rubygems.
Performing inexact rename detection: 100% (22297860/22297860), done.
Update commit message: 310937a546f448603c995fe2bdff0fa7e0610427
...
Your branch is ahead of 'origin/master' by 62 commits.
  (use "git push" to publish your local commits)

You are currently cherry-picking commit 4253d84179d.
  (all conflicts fixed: run "git cherry-pick --continue")
  (use "git cherry-pick --skip" to skip this patch)
  (use "git cherry-pick --abort" to cancel the cherry-pick operation)

nothing to commit, working tree clean
The previous cherry-pick is now empty, possibly due to conflict resolution.
If you wish to commit it anyway, use:

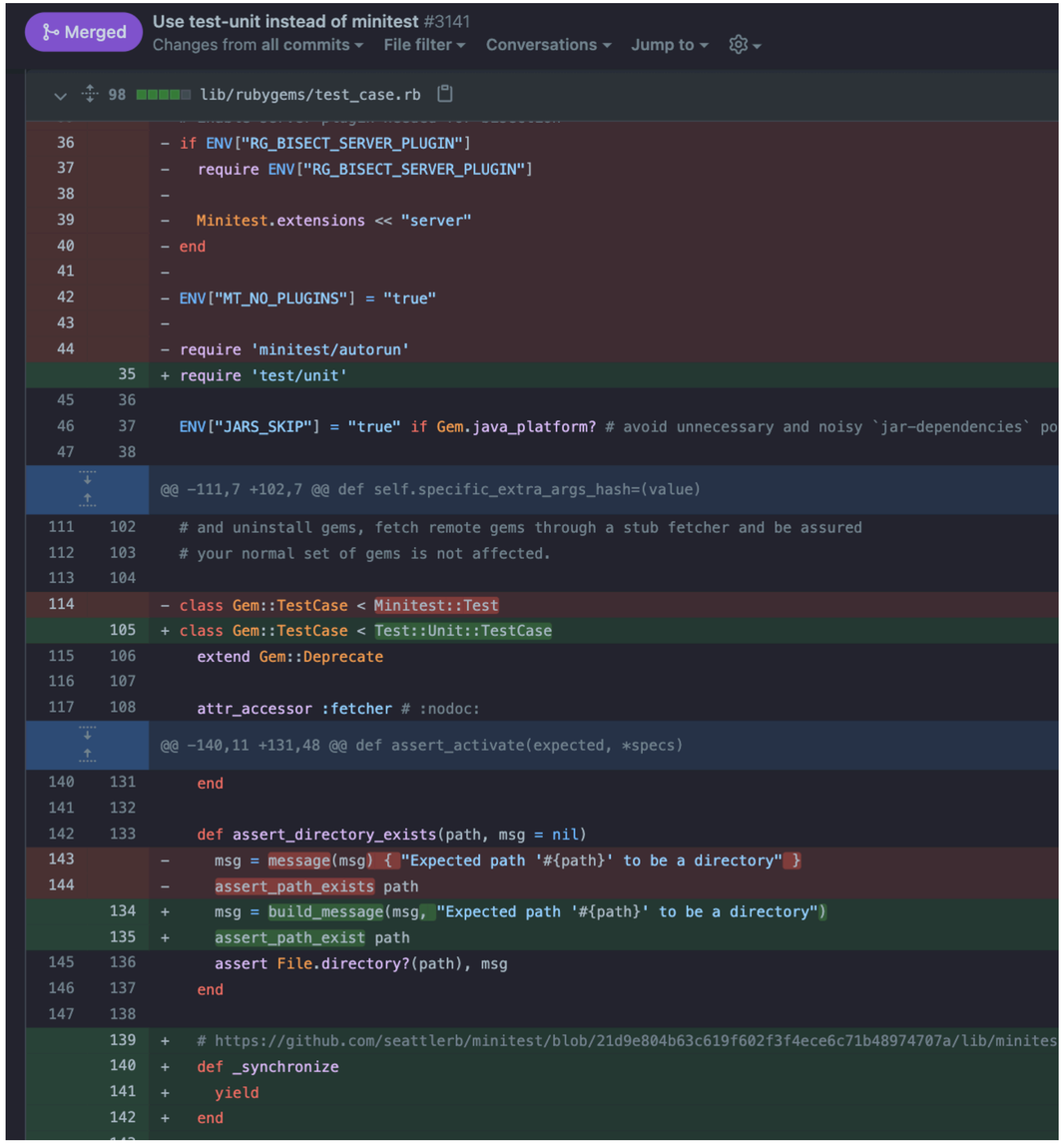
    git commit --allow-empty

Otherwise, please use 'git cherry-pick --skip'
Updated 0 paths from the index
Failed to pick 4253d84179ddee684920087d76b9ab26971a94c9
---- failed commits ----
5c18a8bbb57f4af5e891e5d3935aacc66bf324c7
abb2ae2f591b20d40d8564ec55b748f3bf9dc1f6
e75aa33fe340ffcc21e1d6d87fa548668869c45d
...
5ee72d3cee81ef3ed7145ed638090fed6b007160
4e38926b6e6e4768299cacde742b44169ce183bd
4253d84179ddee684920087d76b9ab26971a94c9

```

Migrate minitest to test-unit

- In the past, We use the custom test framework of Ruby repository for the default gems.
- I migrated them to test-unit + the custom assertions.
- I want to simple mock with test-unit.



What's CoreAssertion?

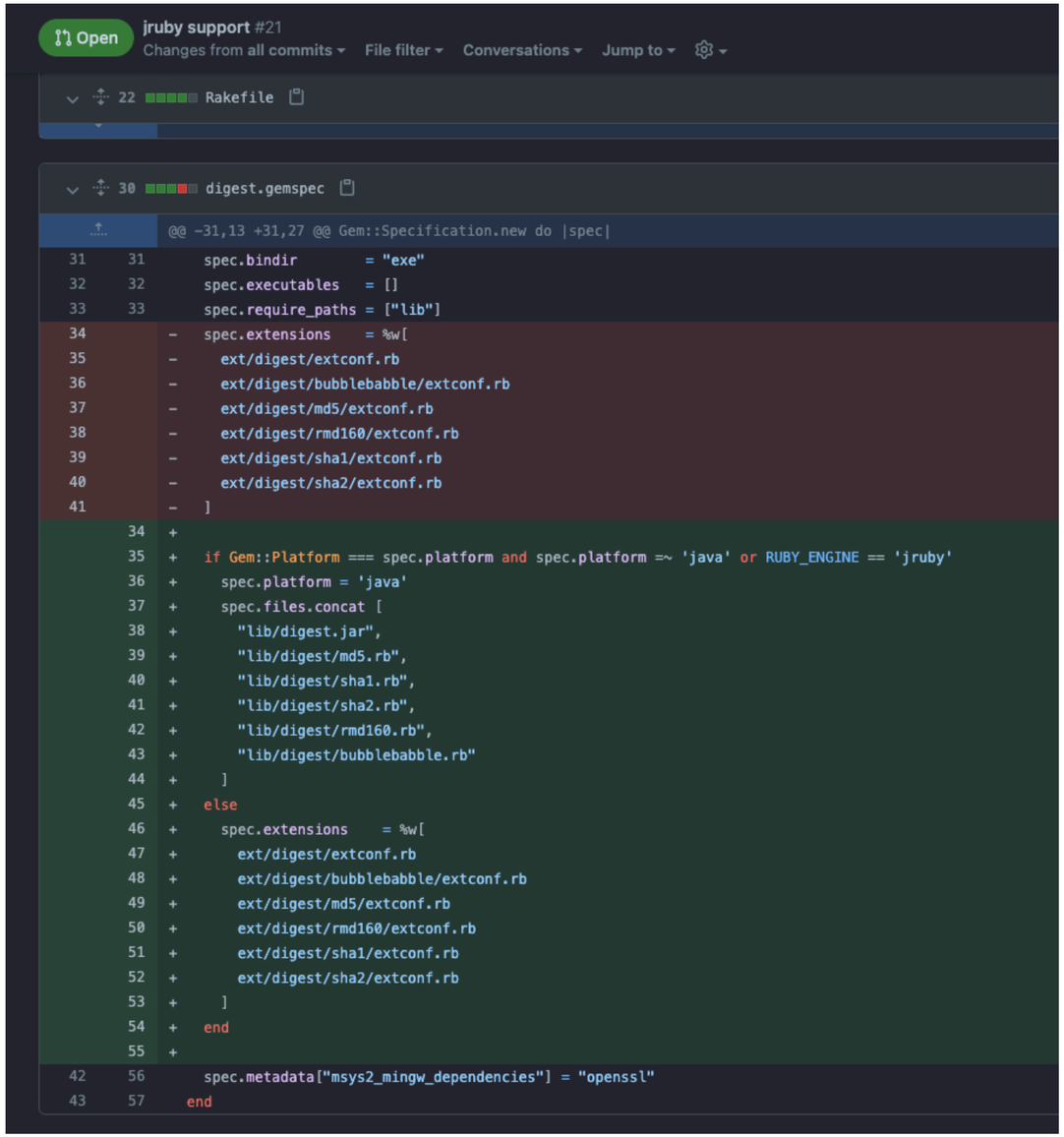
- This provides the advanced usage of ruby interpreter tests
- Integrate envutil
- Add Custom assertions
- Handle test-unit and minitest differences

```
26      module CoreAssertions
27        if defined?(MiniTest)
28          require_relative 'envutil'
29          # for ruby core testing
30          include MiniTest::Assertions
31
32          # Compatibility hack for assert_raise
33          Test::Unit::AssertionFailedError = MiniTest::Assertion
34        else
35          module MiniTest
36            class Assertion < Exception; end
37            class Skip < Assertion; end
38          end
39
40          require 'pp'
41          require_relative 'envutil'
42          include Test::Unit::Assertions
43        end
44
45        def mu_pp(obj) #:nodoc:
46          obj.pretty_inspect.chomp
47        end
48
49        def assert_file
50          AssertFile
51        end
52
53        FailDesc = proc do |status, message = "", out = ""|
54          now = Time.now
55          proc do
56            EnvUtil.failure_description(status, now, message, out)
57          end
58        end
59
60        def assert_in_out_err(args, test_stdin = "", test_stdout = [], test_stderr = [], mess
61                              success: nil, **opt)
```

The feature plan for Ruby 3.1/4.0

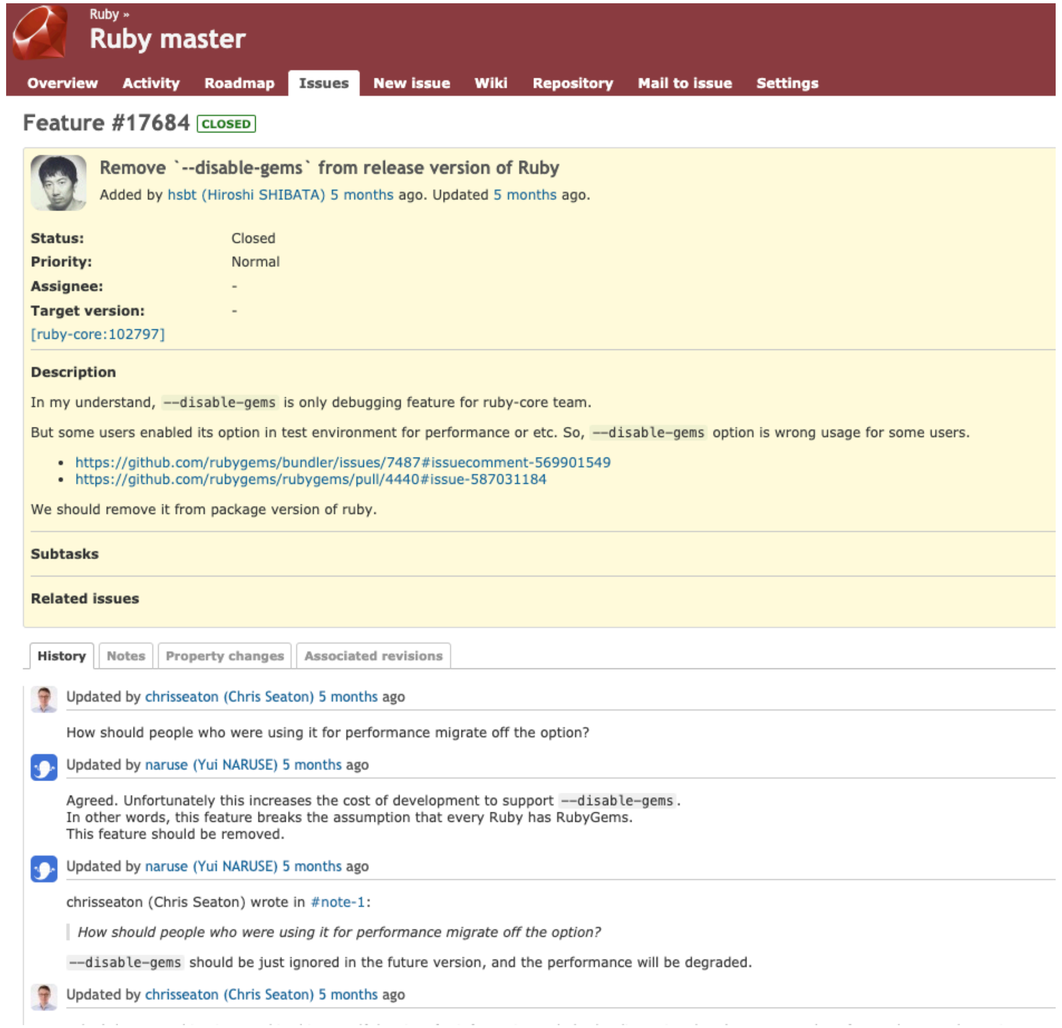
JRuby migration

- Finally, We can migrate JRuby port to the default gems.
- We could care the both specification at one repository.
- But We should coordinate the license of JRuby code.



Avoid to use `--disabled-gems`

- Ruby have a `--disabled-gems` for the Ruby core development.
- We considered to disable/remove it from distributed version because the users misunderstood its usage.
- We should improve the its performance.



Ruby »
Ruby master

Overview | Activity | Roadmap | Issues | New issue | Wiki | Repository | Mail to issue | Settings

Feature #17684 CLOSED

Remove `--disable-gems` from release version of Ruby

Added by hsbt (Hiroshi SHIBATA) 5 months ago. Updated 5 months ago.

Status:	Closed
Priority:	Normal
Assignee:	-
Target version:	-

[ruby-core:102797]

Description

In my understand, `--disable-gems` is only debugging feature for ruby-core team.

But some users enabled its option in test environment for performance or etc. So, `--disable-gems` option is wrong usage for some users.

- https://github.com/rubygems/bundler/issues/7487#issuecomment-569901549
- https://github.com/rubygems/rubygems/pull/4440#issue-587031184

We should remove it from package version of ruby.

Subtasks

Related issues

History | Notes | Property changes | Associated revisions

Updated by chrisseaton (Chris Seaton) 5 months ago

How should people who were using it for performance migrate off the option?

Updated by naruse (Yui NARUSE) 5 months ago

Agreed. Unfortunately this increases the cost of development to support `--disable-gems`.
In other words, this feature breaks the assumption that every Ruby has RubyGems.
This feature should be removed.

Updated by naruse (Yui NARUSE) 5 months ago

chrisseaton (Chris Seaton) wrote in #note-1:

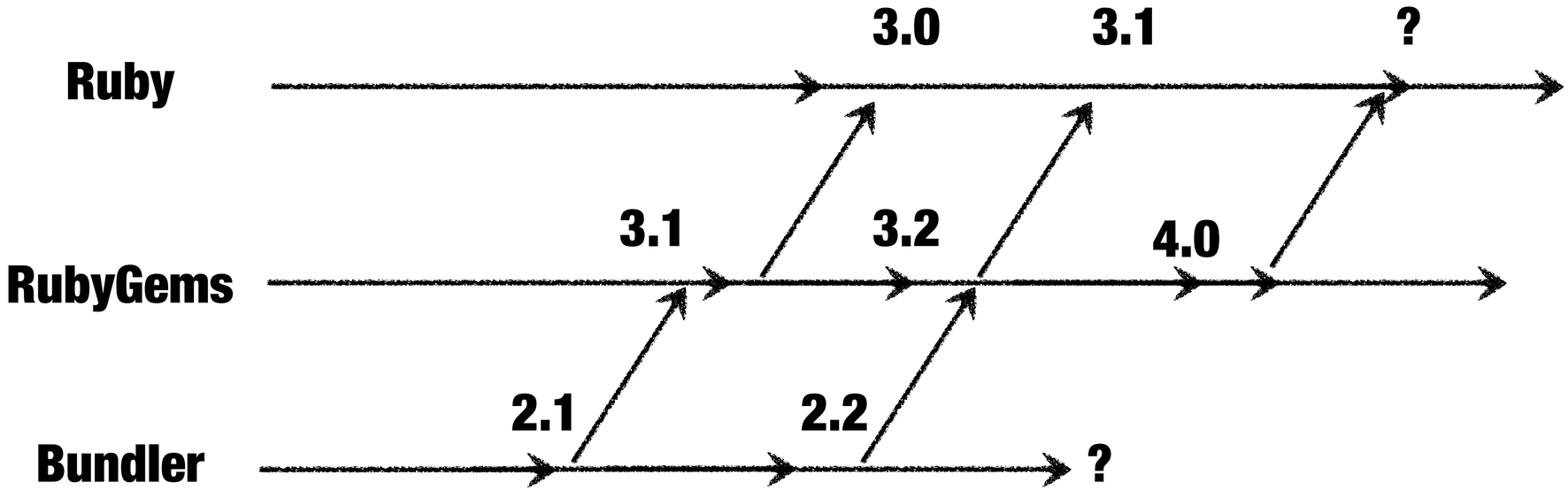
> *How should people who were using it for performance migrate off the option?*

`--disable-gems` should be just ignored in the future version, and the performance will be degraded.

Updated by chrisseaton (Chris Seaton) 5 months ago

Bump up RubyGems/Bundler

We will merge into RubyGems 3.3 and Bundler 2.3 into Ruby 3.1.0-dev. But We have no plan to release RG 4.0 and Bundler 3.0

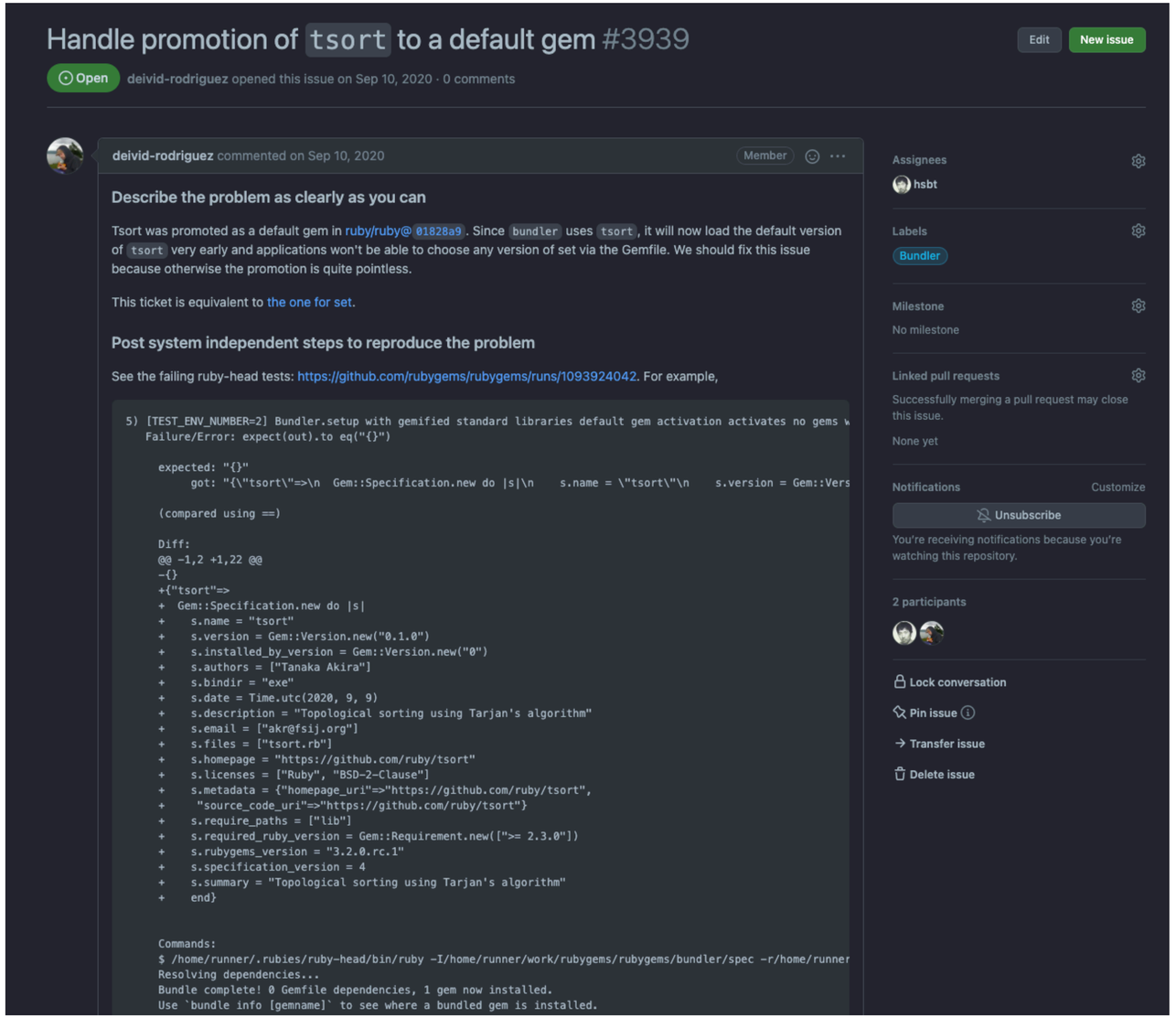


Activation issues about the default gems

We still have a activation issue with RubyGems and the default gems.

- tsort
- ...

We should rewrite it with pure ruby inplementation.



We need the contributors for the default gems

- We need to your help the following topics
  - Ractor support
  - Documentation
- We can give write grant for the default gems repository like ruby/*.
- It’s not ruby-commiter but you can commit to stdlib.



Conclusion

Conclusion

I introduced the following topics.

- The current status of gemification at Ruby 3.1
- The details of developing the default gems
- The future plan for RubyGems/Bundler and Gemification

Let’s join to #rubygems of ruby-jp slack and talk about them.

