

Ruby - Misc #19525

DevMeeting-2023-04-13

03/10/2023 01:48 AM - mame (Yusuke Endoh)

Status:	Closed	
Priority:	Normal	
Assignee:		
Description		
<h2>The next dev meeting</h2> <p>Date: 2023/04/13 13:00-17:00 (JST)</p> <p>Log: https://github.com/ruby/dev-meeting-log/blob/master/2023/DevMeeting-2023-04-11.md</p> <ul style="list-style-type: none">• Dev meeting <i>IS NOT</i> a decision-making place. All decisions should be done at the bug tracker.• Dev meeting is a place we can ask Matz, nobu, nurse and other developers directly.• Matz is a very busy person. Take this opportunity to ask him. If you can not attend, other attendees can ask instead of you (if attendees can understand your issue).• We will write a record of the discussion in the file or to each ticket in English.• All activities are best-effort (keep in mind that most of us are volunteer developers).• The date, time and place of the meeting are scheduled according to when/where we can reserve Matz's time.• <i>DO NOT</i> discuss then on this ticket, please. <h2>Call for agenda items</h2> <p>If you have a ticket that you want matz and committers to discuss, please post it into this ticket in the following format:</p> <pre>* [Ticket ref] Ticket title (your name) * Comment (A summary of the ticket, why you put this ticket here, what point should be discussed, etc.)</pre> <p>Example:</p> <pre>* [Feature #14609] `Kernel#p` without args shows the receiver (kol) * I feel this feature is very useful and some people say :+1: so let discuss this feature.</pre> <ul style="list-style-type: none">• It is recommended to add a comment by 2023/04/10. We hold a preparatory meeting to create an agenda a few days before the dev-meeting.• The format is strict. We'll use this script to automatically create an markdown-style agenda. We may ignore a comment that does not follow the format.• Your comment is mandatory. We cannot read all discussion of the ticket in a limited time. We appreciate it if you could write a short summary and update from a previous discussion.		
Related issues:		
Related to Ruby - Misc #14770: [META] DevelopersMeeting		Open

History

#1 - 03/10/2023 01:48 AM - mame (Yusuke Endoh)

- Related to Misc #14770: [META] DevelopersMeeting added

#2 - 03/28/2023 09:45 AM - hsb (Hiroshi SHIBATA)

- [Bug #19551] Backport commits for CI failures
 - Can I merge PR like "Fix CI failure" for stable branches myself from April 2023?

#3 - 04/03/2023 11:16 AM - byroot (Jean Boussier)

- [Feature #18885] End of boot advisory API (byroot)
 - I'm about to merge the first optimizations, are we still ok with Process.warmup as a name?
- [Feature #19561] ObjectSpace::WeakMap#delete and ObjectSpace::WeakKeyMap#delete (byroot)
 - When used as a weak set, it can be convenient to eagerly delete some keys.
- [Feature #19538] Performance warnings (byroot and eregon)

- Disabled by default, enabled with -W:performance.
- Would allow the VM or YJIT to indicate when it has to deoptimize because of some construct.
- For example: transitioning to SHAPE_TOO_COMPLEX, mega-morphic call sites, etc.
- [Feature [#19528](#)] JSON.load enabling create_additions: true by default is surprising and lead to security vulnerabilities (byroot)
 - JSON.load is the natural method to reach to as "load/dump" is the standard Ruby interface (Marshal.load, YAML.load, etc).
 - Various linters have to ben load and recommend to use parse instead.
 - I think json 2.x should warn when this feature is implicitly enabled, and json 3.x should disable it by default.
 - This has led to multiple CVE in the past.
- [Feature [#19560](#)] IO#close_on_fork= and IO#close_on_fork? (byroot)
 - O_CLOFORK was added to the POSIX spec in 2020, and some system (not Linux) supports it.
 - Ruby could emulate it.
 - Such feature would make it much easier to write fork-safe network clients.

#4 - 04/03/2023 09:14 PM - peterzhu2118 (Peter Zhu)

- [Feature [#19571](#)] Add REMEMBERED_WB_UNPROTECTED_OBJECTS_LIMIT_RATIO to the GC (peterzhu2118)
 - We tested on production traffic on Shopify's [Storefront Renderer](#).
 - We saw a significant improvement in GC time (0.67x on average and 0.49x for p99) and response time (0.96x on average and 0.86x for p99).

#5 - 04/03/2023 10:03 PM - jeremyevans0 (Jeremy Evans)

- [Bug [#4040](#)] SystemStackError with Hash[*a] for Large a (jeremyevans0)
 - [@ko1 \(Koichi Sasada\)](#) fixed this issue for iseq methods accepting splats in Ruby 2.2.
 - We fixed this issue for cfunc methods in January.
 - I found this issue applies to all other method calls.
 - I developed a fix for all cases I could identity (yjit will need updates for it).
 - The fix results in a minor performance decrease in microbenchmarks.
 - I have attempted to offset this performance decrease with major optimizations to bmethod, send, symproc, and method_missing calls, using knowledge gained from fixing this bug.
 - Do we want to fix this issue, and if so, how much of a performance loss are we willing to accept?
 - I propose we should accept this fix if it doesn't result in more than 0.2% performance decrease in production benchmarks such as railsbench.
- [Bug [#19533](#)] Behavior of ===/include? on a beginless/endless range (nil..nil) changed in ruby 3.2 (jeremyevans0)
 - Do we want to make beginless+endless Range#include? return true for all objects?
 - A decision was made in August 2022 to have include? raise exception for beginless/endless ranges.
 - Looking back, it's possible that it was desired for the beginless^endless case and not the beginless+endless case.
- [Bug [#19564](#)] Range.cover? fails for Range wrapped in SimpleDelegator (jeremyevans0)
 - Do we want to handle delegate objects in Range#cover??
 - If so, do we want to add Range#to_range and have Range#cover? check for that method and call it if it exists?
- [Feature [#19573](#)] Add Class#singleton_inherited (jeremyevans0)
 - This would be similar to Class#inherited, but called for singleton class creation instead of normal subclass creation.
 - This could be used to warn or raise on singleton class creation, or modify the object to change behavior.

#6 - 04/12/2023 02:41 PM - matz (Yukihiro Matsumoto)

- [Bug [#19246](#)] Rebuilding the loaded feature index much slower in Ruby 3.1
 - Personally asked for the situation.

#7 - 04/13/2023 12:16 AM - palkan (Vladimir Dementyev)

- [Feature [#19591](#)] Add symbolize_keys to MatchData#named_captures
 - Looking for Matz's feedback

#8 - 04/14/2023 02:35 AM - mame (Yusuke Endoh)

- Description updated

- Status changed from Open to Closed