The next dev meeting

Date: 2021/01/13 13:00-17:00
Place/Sign-up/Agenda/Log: https://github.com/ruby/dev-meeting-log/blob/master/DevelopersMeeting20210113Japan.md

- Dev meeting IS NOT 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 log about the discussion to a 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 are scheduled according to when/where we can reserve Matz's time.
- DO NOT discuss then on this ticket, please.

Call for agenda items

If you have a ticket that you want matz and committers to discuss, please post it into this ticket in the following format:

- [Ticket ref] Ticket title (your name)
  * Comment (A summary of the ticket, why you put this ticket here, what point should be discussed, etc.)

Example:

- [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.

- Comment deadline: 2021/01/06 (one week before the 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.

Related issues:

Related to Ruby master - Misc #14770: [META] DevelopersMeeting

History

#1 - 12/27/2020 10:17 AM - mame (Yusuke Endoh)
- Related to Misc #14770: [META] DevelopersMeeting added

#2 - 12/27/2020 11:08 AM - mame (Yusuke Endoh)

- [Feature #17472] HashWithIndifferentAccess like Hash extension (mame)
  * It is a good idea for Ruby 3.1 to provide small but immediate improvements like this. The constant name should be discussed.
- [Feature #17473] Make Pathname to embedded class of Ruby (mame)
  * We want to make Rubygems independent with any gems. Rubygems heavily depends on Pathname. So how about making it built-in?
- [Feature #16989] Sets: need ♥️ (mame)
  * SortedSet has been removed from set.rb, so there is no longer a dependency problem.
- [Feature #17479] Enable to get "--backtrace-limit" value by "$-B" (mame)
  * Some libraries need to know what --backtrace-limit is specified. How about providing a command-line option -B as an alias, and a special variable $-B too?

#3 - 12/27/2020 12:42 PM - chrisseaton (Chris Seaton)
- [Feature #17327] The Queue constructor should take an initial set of items (chrisseaton)
  This simple feature captures a common pattern. There's a PR with an implementation based on initial feedback and specs
  [https://github.com/ruby/ruby/pull/3768](https://github.com/ruby/ruby/pull/3768).

#4 - 12/30/2020 05:49 AM - k0kubun (Takashi Kokubun)

- [Feature #16806] Struct#initialize accepts keyword arguments too by default (k0kubun)
  Is there any update on kwargs since the last discussion? Can we introduce a warning for it in 3.1?
- [Feature #17490] Rename RubyVM::MJIT to RubyVM::JIT (k0kubun)
  Any objections, or suggestions to the release plan?

#5 - 01/01/2021 12:18 PM - zverok (Victor Shepelev)

- [Misc #17499] Documentation backporting (zverok)
  Address systematic lack of docs at docs.ruby-lang.org
- [Bug #16497] String#internal_encoding is broken (more severely in 2.7) (zverok)
  Still is, as of 3.0.0
- [Feature #17407] Fiber.current and require 'fiber' (zverok)
  The requirement seem to not be rethought for a long time, can it be lifted?
- [Feature #17312] New methods in Enumerable and Enumerator: Lazy (zverok)
  Matz approved #compact, PR is provided, can it be merged?..
- [Feature #17330] Object#non (zverok)
  Experimental proposal about syntax like calculation.non(!&zero?) || something
- [Feature #17496] Add constant Math::TAU (jzakiya)
  Reconsider adding to Ruby core, as more languages have. Increased adoption/use since last consideration. It has become "time proven",
  [https://bugs.ruby-lang.org/issues/4897#note-41](https://bugs.ruby-lang.org/issues/4897#note-41)

#6 - 01/01/2021 12:50 PM - Eregon (Benoit Daloze)

- [Feature #17500] Move RubyVM:* to ExperimentalFeatures (egoron)
  I think this is the best solution, for everyone. Could matz and ruby-core agree to this?

#7 - 01/02/2021 02:37 AM - jzakiya (Jabari Zakiya)

- [Feature #17496] Add constant Math::TAU (jzakiya)
  Reconsider adding to Ruby core, as more languages have. Increased adoption/use since last consideration. It has become "time proven",
  [https://bugs.ruby-lang.org/issues/4897#note-41](https://bugs.ruby-lang.org/issues/4897#note-41)

#8 - 01/02/2021 06:56 AM - marcandre (Marc-Andre Lafortune)

- [Misc #0] Comment Deadline (marcandre)
  Please make comment deadline closer to meetings (1 or 2 days, not 7).

#9 - 01/03/2021 03:03 PM - nobu (Nobuyoshi Nakada)

- [Feature #12607] Ruby needs an atomic integer (shyouhei (Shyouhei Urabe))
  dummy comment
- [Feature #17099] Remove boolean argument and warning from Module#attr (S_H_)
  It has been warned since 1.9.1.
- [Feature #17485] Keyword argument for timezone in Time.new (nobo (Nobuyoshi Nakada))
  Let Time.new accept a timezone as a keyword argument, as well as Time.at and Time.now.
- [Bug #17504] Allow UTC offset without colons per ISO-8601 (nobo (Nobuyoshi Nakada))
  Although ISO-8601 allows UTC offset to be +HHMM as well as +HH:MM, Time.new, Time#tolocal etc reject that format as invalid.
- [Bug #17509] Custom respond_to? methods in modules break defined?(super) (benediktdeicke (Benedikt Deicke))
  Custom respond_to? can't know the point to continue searching the method entry.
- [Bug #17523] Inconsistent Warning[] values in scripts loaded by -r option (nobo (Nobuyoshi Nakada))
  While -w option affects $VERBOSE for both the main and required scripts, but affects Warning[] values only in the main script.

#10 - 01/06/2021 01:50 AM - jeremyevans0 (Jeremy Evans)

- [Bug #17428] Prohibit include/prepend in refinement modules (jeremyevans0)
  I've added a pull request for this, is it OK to merge?
  Do we want to implement a feature to copy methods from a module into a refinement, such as the :import option?
- [Bug #17423] Prepend should prepend a module before the class (jeremyevans0)
  Should Module#prepend add iclasses to the start of the ancestor chain if the receiver already includes the module?
  Should Module#prepend add iclasses to the start of the ancestor chain if the receiver has already prepended the module?
  Should Module#prepend add iclasses to the start of the ancestor chain if the ancestor chain already starts with the iclasses for the
  prepended module?
  If the answer to any question is yes, can we decide on details for how prepend should work?

03/17/2022 2/3
[Bug #17517] File.expand_path returns us-ascii when both arguments are ascii compat (znz)
require_relative use expand_path, and it returns us-ascii

#12 - 01/12/2021 09:02 AM - mame (Yusuke Endoh)
marcandre (Marc-Andre Lafortune) wrote in #note-8:

- [Misc #0] Comment Deadline (marcandre)
  - Please make comment deadline closer to meetings (1 or 2 days, not 7).

Okay. In the next ticket, I will change it to 3 days and rephrase "deadline" to "recommendation".

The reason of the deadline is that a few voluntary committers (including I) hold a preparatory meeting to preread the tickets and create an agenda a few days before the dev-meeting. It is recommended to add a ticket before the pre-reading meeting.

So far, many of the pre-reading meetings were held two days before the dev-meeting (in JST). Considering a time difference, I'd like to set it to 3 days before. You can add comments after the deadline, but sorry if I miss them.

#13 - 01/13/2021 04:26 PM - mame (Yusuke Endoh)
- Status changed from Open to Closed
- Description updated