The next dev meeting

Date: 2021/10/21 13:00-17:00 +09:00
Place/Sign-up/Agenda/Log: https://github.com/ruby/dev-meeting-log/blob/master/DevelopersMeeting20211021Japan.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.

- It is recommended to add a comment by 2021/10/18. 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 master - Misc #14770: [META] DevelopersMeeting Open
- **[Feature #18083]** Capture error in ensure block.
  - Can we deprecate $! (& $@)? It's dangerous and every use case is better with explicit capture. Also may have some performance advantages.

- **[Feature #18020]** Introduce IO::Buffer for fiber scheduler.
  - For efficient file IO we need an internal buffer with well defined semantics suitable for IO.

- **[Feature #17369]** Introduce non-blocking Process.wait, Kernel.system and related methods.
  - Currently, $? is thread-local, but it should be fiber local.
  - Should we try to fix $? or deprecate it?
  - Introduce new interface? system(...) {status} ...

- **[Feature #18194]** No easy way to format exception messages per thread/fiber scheduler context.
  - Can we come to an agreement on what this looks like for Ruby 3.1?
  - Can we introduce fiber scheduler hook for unhandled exceptions?

---

**#4 - 09/28/2021 11:26 PM - shugo (Shugo Maeda)**

- **[Bug #17429]** Prohibit include/prepend in refinement modules
  - Is the name Refinement#import OK?

**#5 - 10/04/2021 11:52 PM - duerst (Martin Dürst)**

- **[Feature #18239]** Variable Width Allocation: Strings
  - Discuss deployment schedule and binary incompatibility issues.

**#6 - 10/09/2021 07:58 AM - nobu (Nobuyoshi Nakada)**

- **[Feature #18242]** Allow multiple assignment in logical expression
  - 1 < 2 and a, b = 2, 1 and so on.

**#7 - 10/11/2021 09:57 PM - jeremyevans0 (Jeremy Evans)**

- **[Misc #18248]** Add Feature Triaging Guide (jeremyevans0)
  - Is it OK to add a feature triaging guide and start using it triage feature requests?

- **[Bug #18066]** Load did_you_mean/error_highlight even with --disable-gems (jeremyevans0)
  - Should --disable-gems also disable did_you_mean and error_highlight?
  - Most developers appear in favor, though mame (Yusuke Endoh) is against (but not strongly opposed).

- **[Bug #18187]** Float#clamp() returns ArgumentError (comparison of Float with 1 failed) (jeremyevans0)
  - How should Float#clamp handle case where receiver is NaN?
  - Currently it raises ArgumentError, but that is due to implementation details, not by design.
  - Is it OK to change it to return the receiver (NaN) in this case?

- **[Bug #12436]** newline argument of File.open seems not respected on Windows (jeremyevans0)
  - The option is currently ignored due to an oversight when the code was originally added.
  - Is it OK to introduce lf_newline transcoder and newline: :lf option to fix this?
  - Also, is newline: :crlf supposed to convert \r\n to \n during read only on Windows?

**#8 - 10/14/2021 02:35 PM - byroot (Jean Boussier)**

- **[Feature #10917]** Add GC.stat[:total_time] when GC profiling enabled
  - ko1 (Koichi Sasada) did submit a pull request which seem fine except maybe for 32bits support.
  - ko1 (Koichi Sasada) did also measure the overhead and it appeared small.
  - Could this feature be merged for Ruby 3.1?

**#9 - 10/15/2021 11:09 AM - byroot (Jean Boussier)**

- **[Feature #17760]** Where we should install a header file when gem install --user?
  - digest have been gemified. However its extconf.rb have $INSTALLFILES = { "digest.h" => "$(HDRDIR)" } which fail for most people because ruby's include directory is owned by root.
  - What should rubygems do here? Have a secondary user owned include directory?
  - If I have several versions of digest installed? How gems linking against it would know which version to use?
  - If there's no clear solution, with 3.1.0 approaching, should we revert that digest gemification so that 3.1.0 is actually usable?

**#10 - 10/18/2021 03:21 AM - nobu (Nobuyoshi Nakada)**

  - Description updated

**#11 - 10/19/2021 09:28 AM - Eregon (Benoit Daloze)**

- **[Feature #18256]** Change the canonical name of Thread::Mutex, Thread::Queue, Thread::SizedQueue and Thread::ConditionVariable to just Mutex, Queue, SizedQueue and ConditionVariable (eregon)
OK? (sorry for late addition for the meeting, we can move to next meeting if needed, but would be best to decide sooner than later)

#12 - 10/20/2021 01:28 AM - Dan0042 (Daniel DeLorme)
- [Feature #17795] Around Process.fork callbacks API (dan0042)
  - A few naming suggestions have been made; does Matz like any of them?
- [Feature #17837] Add support for Regexp timeouts (dan0042)
  - Adding this to ruby 3.1 with a safe (high) value for Regexp.backtrack_limit would be an improvement on the current lack of limit, and would allow to get measurements from real-world applications for a better default value.

#13 - 10/20/2021 08:25 AM - byroot (Jean Boussier)
- [Feature #18254] Add an offset parameter to String#unpack and String#unpack1
  - Binary protocol parsers currently have to slice their buffer repeatedly, this new argument would save a lot of string copying.

#14 - 10/20/2021 02:25 PM - Eregon (Benoit Daloze)
- [Bug #17146] Queue operations are allowed after it is frozen. (eregon)
  - This is an obvious inconsistency and it makes no sense whenever we discuss deep freezing object graphs. Let's fix it.

#15 - 10/21/2021 11:37 AM - mame (Yusuke Endoh)
We held the dev-meeting but didn't finish until the last topic. An extra meeting will be held at the next Monday.

#16 - 10/25/2021 09:11 AM - mame (Yusuke Endoh)
- Status changed from Open to Closed
- Description updated