The next dev meeting

Date: 2020/02/27 13:00-17:00
Place/Sign-up/Agenda/Log: https://docs.google.com/document/d/1IqCMi27w7KYeCpke6-dClPV3TLH2QbnDDw5TNOeisc

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 (ko1)
  * I feel this feature is very useful and some people say :+1: so let discuss this feature.

Comment deadline: 2020/02/20 (one week before the meeting)
The format is strict. We'll use this script to automatically create a 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 Open

History

#1 - 01/26/2020 02:12 AM - mame (Yusuke Endoh)
- Related to Misc #14770: [META] DevelopersMeeting added

#2 - 01/26/2020 02:13 AM - mame (Yusuke Endoh)
- Description updated

#3 - 01/26/2020 04:12 PM - znz (Kazuhiro NISHIYAMA)
- [Misc #16483] How about stopping new *.tar.bz2 releases? (znz)
  * Stop new bz2 releases, keep already existing bz2 releases.

#4 - 01/27/2020 03:56 AM - jeremyevans0 (Jeremy Evans)
- [Feature #9573] descendants of a module don't gain its future ancestors, but descendants of a class, do (jeremyevans0)
  * A patch exists for Module#include that passes make check, is it OK to merge?
  * A patch exists for Module#prepend that still has many issues to resolve and requires creating origin classes for almost all modules, do we
want to try to support this feature for Module#prepend?

- [Feature #13675] Should Zlib::GzipReader#ungetc accept nil? (jeremyevans0)
  - Has matz decided whether to fix IO#ungetc? If so, is the patch OK?
- [Feature #11304] [PATCH] Kernel.global_variables should observe $~ (jeremyevans0)
  - Do we want to modify the behavior of Kernel.global_variables for $1, $2, etc.
  - Do we want defined? for $& $' $+ $1 $2 to depend on whether $~ is set?
  - Should defined? for global variable aliases always be the same as what they alias?

**#5 - 01/27/2020 09:39 AM - byroot (Jean Boussier)**

- [Feature #16557] Deduplicate Regexp literals (byroot)
  - #16377 made literal regexps frozen, so they can be deduplicated without further backward compatibility change
  - The estimated memory saving is about 1.4 MB on Redmine (and pretty much all Rails apps).

**#6 - 02/04/2020 10:36 AM - sawa (Tsuyoshi Sawada)**

- [Feature #15722] Kernel#case? (sawa)
  - "foo".case?(Symbol, String) # => true

**#7 - 02/11/2020 11:02 PM - kirs (Kir Shatrov)**

- [Feature #16476] Socket.getaddrinfo cannot be interrupted by Timeout.timeout (kirs)
  - Would we be happy with the proposed solution leveraging getaddrinfo_a? If yes, is it good to merge?

**#8 - 02/18/2020 09:19 PM - Eregon (Benoit Daloz)

- [Feature #16463] Fixing *args-delegation in Ruby 2.7: ruby2_keywords semantics by default in 2.7.1 (eregon)
  - It would be good to decide.
  - We can also discuss it at the RubyKaigi dev meeting, but is that not too late for 2.7.1?

**#9 - 02/19/2020 05:57 PM - Dan0042 (Daniel DeLorme)**

- [Feature #16511] Staged warnings and better compatibility for keyword arguments in 2.7.1 (Dan0042)
  - All the benefits of separation of keyword/positional arguments.
  - Less disruptive migration in the near term; only the strict necessary breaks next year, fewer gems to upgrade.
  - Fewer incompatibilities, inconsistencies, and side effects. (specifics in note 12)
  - For some reason it seems no one understands my proposal? I can volunteer to answer questions at the dev meeting... if desired.

[Note 12](https://git.io/JvWXA) (last 4 commits; other commits with "refactor" are not strictly relevant)
  - The implementation touches a lot of code; it's well-tested, but maybe too much, too late.

02/22/2020