

## Ruby master - Misc #16775

### DevelopersMeeting20200514Japan

04/10/2020 02:47 PM - mame (Yusuke Endoh)

<b>Status:</b> Closed	
<b>Priority:</b> Normal	
<b>Assignee:</b>	
<b>Description</b>	
<h2>The next dev meeting</h2>	
<b>Date:</b> 2020/05/14 13:00-17:00	
Place/Sign-up: <a href="https://docs.google.com/document/d/11rxl64ipKrYIDDsGZBfqjYHNepaOWLv1DZBYAdVHFF8/edit">https://docs.google.com/document/d/11rxl64ipKrYIDDsGZBfqjYHNepaOWLv1DZBYAdVHFF8/edit</a>	
Agenda/Log: <a href="https://github.com/ruby/dev-meeting-log/blob/master/DevelopersMeeting20200514Japan.md">https://github.com/ruby/dev-meeting-log/blob/master/DevelopersMeeting20200514Japan.md</a>	
<ul style="list-style-type: none"><li>• Dev meeting <i>IS NOT</i> a decision-making place. All decisions should be done at the bug tracker.</li><li>• Dev meeting is a place we can ask Matz, nobu, nurse and other developers directly.</li><li>• 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).</li><li>• We will write a log about the discussion to a file or to each ticket in English.</li><li>• All activities are best-effort (keep in mind that most of us are volunteer developers).</li><li>• The date, time and place are scheduled according to when/where we can reserve Matz's time.</li><li>• <i>DO NOT</i> discuss then on this ticket, please.</li></ul>	
<h2>Call for agenda items</h2>	
If you have a ticket that you want matz and committers to discuss, please post it into this ticket in the following format:	
<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>	
Example:	
<pre>* [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.</pre>	
<ul style="list-style-type: none"><li>• Comment deadline: 2020/05/07 (one week before the meeting)</li><li>• The format is strict. We'll use <a href="#">this script to automatically create an markdown-style agenda</a>. We may ignore a comment that does not follow the format.</li><li>• Your comment is mandatory. We cannot read all discussion of the ticket in a limited time.</li></ul>	
<b>Related issues:</b>	
Related to Ruby master - Misc #14770: [META] DevelopersMeeting	<b>Open</b>

## History

### #1 - 04/10/2020 02:47 PM - mame (Yusuke Endoh)

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

### #2 - 04/10/2020 02:50 PM - mame (Yusuke Endoh)

- Description updated

### #3 - 04/18/2020 08:22 PM - greggzst (Grzegorz Jakubiak)

- [Misc #16778] Should we stop vendoring default gems code? (greggzst)
  - it makes maintenance easier and removes the need for synchronization
  - removes confusion for external contributors
  - Ruby uses git so git submodules could be used
  - it's possible to setup CI to run against Ruby master

### #4 - 04/23/2020 03:41 PM - zverok (Victor Shepelev)

- [Feature [#16812](#)] Allow slicing arrays with ArithmeticSequence (zverok)
  - `ary[(5..20) % 2]`—each second element between 5 and 20
  - `ary[(0..) % 3]`—each third element
  - `ary[10.step(by: -1)]`—elements 10, 9, 8, 7 ....
  - Python has and uses it (in form of `beg:end:step`), and ArithmeticSequence and Range#% were introduced for this

#### #5 - 04/24/2020 02:11 AM - ioquatix (Samuel Williams)

- [Feature [#16815](#)] Introduce Fiber#backtrace. (ioquatix)
  - Add Fiber#backtrace which gets any fiber backtrace. Very useful for debugging fiber state.
- [Feature [#16786](#)] Light weight scheduler for improved concurrency. (ioquatix)
  - Please check it and confirm if we can merge it, or outline what changes are required.

#### #6 - 04/24/2020 05:13 AM - k0kubun (Takashi Kokubun)

- [Feature [#16806](#)] Struct#initialize accepts keyword arguments too by default (k0kubun)
  - This obviates `keyword_init: true`. Is the described incompatibility and release plan (3.0: warn, 3.1: introduce) acceptable?

#### #7 - 04/30/2020 10:54 AM - zverok (Victor Shepelev)

- [Feature [#16822](#)] Array slicing: nils and edge cases (zverok)
  - Never return nil from `ary[start...end]` even if out of arrays' bounds

#### #8 - 04/30/2020 07:18 PM - stan3 (Tristan Hill)

- [Feature [#9758](#)] Allow setting SSLContext#extra\_chain\_cert in Net::HTTP (stan3)
  - useful to allow https with cert chain, small patch, few +1s

#### #9 - 05/02/2020 02:54 AM - shyouhei (Shyouhei Urabe)

- [Misc [#16747](#)] Repository reorganization request (shyouhei)
  - [hsbt \(Hiroshi SHIBATA\)](#) says:
 

I remember [nobu \(Nobuyoshi Nakada\)](#) has the working branch about this.

I would like to know its current situation.

#### #10 - 05/02/2020 09:15 AM - sawa (Tsuyoshi Sawada)

- [Feature [#15771](#)] Add String#split option to set split\_type string with a single space separator (sawa)
  - Proposal by 284km. Allow splitting literally by " " when the argument is " ".

#### #11 - 05/06/2020 11:40 AM - byroot (Jean Boussier)

- [Feature [#16832](#)] Use #name rather than #inspect to build "uninitialized constant" error messages (byroot)
  - As #name on Module and Class is more likely to return a proper constant name than #inspect

#### #12 - 05/07/2020 07:52 AM - mame (Yusuke Endoh)

- [Feature [#8661](#)] Add option to print backtrace in reverse order (stack frames first and error last) (mame)
  - matz, could you check the spec of `--suppress-backtrace`? And should we change the name to `--backtrace-limit` or something?

#### #13 - 05/07/2020 08:15 AM - k0kubun (Takashi Kokubun)

- [Feature [#16837](#)] Can we make Ruby 3.0 as fast as Ruby 2.7 with the new assertions? (k0kubun)
  - Do we have a policy on what kind of assertions should be always enabled and what should be CI-only, especially when it has impact on performance?

#### #14 - 05/07/2020 08:31 AM - mrkn (Kenta Murata)

- [Feature #16827] C API for writing custom random number generator that can be used as Random objects (mrkn)
  - I need such a feature to implement custom RNGs in an extension library

#### #15 - 05/07/2020 05:03 PM - Eregon (Benoit Daloze)

- [Feature #16254] MRI internal: Define built-in classes in Ruby with `__intrinsic__` syntax (eregon)
  - Thoughts on using `Primitive.name(a, b)` instead of `__builtin_name(a, b)`?
  - I think this is an important opportunity to share more code between implementations.

#### #16 - 05/07/2020 05:36 PM - Eregon (Benoit Daloze)

- [Feature #15921] R-assign (rightward-assignment) operator (eregon)
  - Could matz and other committers clarify the motivation to introduce this? There is no pipeline operator currently so it seems of limited usage.
  - Changing syntax often divides the community (e.g., <https://twitter.com/devoncestes/status/125622228431228933>), should we be more careful when introducing new syntax? For instance, on syntax issues matz considers to merge, he could state his intention, tweet about the proposal, let it be for at least a month and then decide (merge or not). A blog post by ruby core would be another way to trigger feedback before merging. Often it feels like a decision "out of the blue" with no clear motivation. Discussion after merging feels suboptimal because people realize they have very little chance to change anything and so just love or hate it. Being in master doesn't help much because most people won't try it, yet they can share their opinion based on code snippets using the new syntax.
- [Feature #16746] Endless method definition (eregon)
  - Is it intended to allow multi-line definitions? Seems redundant and likely to make code less readable (end for class/module/do/ff/while/... still exist).

#### #17 - 05/09/2020 07:07 AM - mame (Yusuke Endoh)

- [Feature #16828] Introduce find patterns (ktsj)
  - I implemented it as matz wanted it. Please discuss it on the dev-meeting.

#### #18 - 05/11/2020 12:55 AM - ko1 (Koichi Sasada)

- [Misc #16803] Discussion: those internal macros reside in public API headers (ko1)
  - confirm the discussion.

#### #19 - 05/11/2020 11:42 AM - byroot (Jean Boussier)

- [Feature #16847] Cache instruction sequences by default (byroot)
  - This make code loading about 30% faster.
  - All it's needed is to agree on where and how to store the cache.
- [Feature #16848] Allow callables in `$LOAD_PATH` (byroot)
  - They would receive the relative path as first argument, and be expected to return an absolute path or nil
  - This would make implementing various lookup caching strategies in Rubygems/Bundler much easier and more accurate.
  - Such caching can yield massive boot time improvements.

#### #20 - 05/16/2020 08:18 AM - mame (Yusuke Endoh)

- Description updated

#### #21 - 05/16/2020 08:20 AM - mame (Yusuke Endoh)

We've hold a dev-meeting at 14th, but could go through only half topics. We'll hold an extra meeting at 26th.

Log: <https://github.com/ruby/dev-meeting-log/blob/master/DevelopersMeeting20200514Japan.md>

#### #22 - 05/18/2020 05:32 PM - shevegen (Robert A. Heiler)

mame wrote:

[...] We'll hold an extra meeting at 26th.

\o/

#### #23 - 06/04/2020 01:24 AM - mame (Yusuke Endoh)

- Status changed from Open to Closed