Ruby trunk - Misc #14698

DevelopersMeeting20180517Japan

04/19/2018 06:59 AM - ko1 (Koichi Sasada)

Status:	Closed
Priority:	Normal
Assignee:	matz (Yukihiro Matsumoto)

Description

Date: 2018/05/17 (Thu)
Time: 14:00-18:00 (JST)
Place: Cookpad Inc. (Tokyo, Japan)
Sign-up: https://ruby.connpass.com/event/85917/
Log:
https://docs.google.com/document/d/e/2PACX-1vR2LdBE87iEcEsVuUUr0G2L6LxSPeGMg_0oeHeh0HYmX36iIa9zkWYlFHilH5D4I_RBJpQnr09yOZaE/pub

Please comment your favorite ticket numbers you want to ask to discuss with your *SHORT* comment or summary.
(your summary/comment will help us because we don't need to read all of ticket comments)

DO NOT discuss then on this ticket, please.

From this time, we use a ticket to make dev-meeting agenda page instead of a wiki page https://bugs.ruby-lang.org/projects/ruby/wiki
.

NOTE

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 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 is scheduled according to when/where we can reserve Matz's time.

Agenda

Next dev-meeting

https://bugs.ruby-lang.org/issues/14769

About 2.6 timeframe

preview2 will be released RubyKaigi 2018 day 1 or 2

From Attendees

- [Bug #14699] Subtle behaviors with endless range (mame)
- [Feature #14724] chains of inequalities (Martin)
  - Proposal by gotoken (Kentaro Goto) to allow 0 <= a < 10 as a shortcut of 0<=a && a<10, and so on; patch by Nobu avaliable

(will be edited later)
(if you have a write access permission, please list directly)

From non-attendees

- [Feature #14697] Introducing Range#% as an alias to Range#step (mrkn)
  - Matz already commented LGTM. Please judge whether this is acceptable.
- [Feature #14724] chains of inequations (mrkn)
  - This new syntax can also reduce the duplicated evaluations of common terms, such as timeval.tv_sec in the descriptiohn

of the proposal.

Functional programming: (zverok)

- [Feature #6284] Add composition for procs
    - 6-year-old proposal. Matz: "Positive about adding function composition. But we need method name consensus before adding it? Is #* OK for everyone?"
- [Feature #13581] Syntax sugar for method reference
    - 1-year-old proposal. Matz: "I am for adding syntax sugar for method reference. But I don't like proposed syntax (e.g. ->). Any other idea?"
- [Feature #11161] Proc/Method#rcurry working like curry but in reverse order
    - 3-year-old proposal with absolutely no reaction. I've added real-life examples there several months ago.
- [Feature #14390] UnboundMethod#to_proc
- [Feature #14423] Enumerator from single object (Object#enumerate)

Misc:

- [Bug #14575] Switch Range#=== to use cover? instead of include? (zverok)
- [Feature #14473] Add Range#subrange? (greggzst)
- [Feature #14097] Add union and difference to Array (ana06)
- [Feature #14105] Introduce xor as alias for Set#^ (ana06)
- [Misc #14760] cross-thread IO#close semantics (normal)

Docs (probably not to discuss on development meeting, but I am not sure where I can post it to have it more visible) (zverok):

- Method: https://bugs.ruby-lang.org/issues/14483
- Proc: https://bugs.ruby-lang.org/issues/14610
- MatchData: https://bugs.ruby-lang.org/issues/14450
- yield_self: https://bugs.ruby-lang.org/issues/14436

(will be edited later)
(if you have a write access, please list directly)

Related issues:	
Related to Ruby trunk - Misc #14770: [META] DevelopersMeeting	**Open**

History

#1 - 04/19/2018 07:01 AM - mrkn (Kenta Murata)

- Description updated

#2 - 04/19/2018 07:03 AM - ko1 (Koichi Sasada)

- Description updated

#3 - 04/19/2018 07:05 AM - mrkn (Kenta Murata)

- Description updated

#4 - 04/19/2018 07:26 AM - zverok (Victor Shepelev)

your favorite ticket numbers you want to ask to discuss with your SHORT comment or summary.

Functional programming:

- [Feature #6284] Add composition for procs
    - 6-year-old proposal. Matz: "Positive about adding function composition. But we need method name consensus before adding it? Is #* OK for everyone?"
- [Feature #13581] Syntax sugar for method reference
    - 1-year-old proposal. Matz: "I am for adding syntax sugar for method reference. But I don't like proposed syntax (e.g. ->). Any other idea?"
- [Feature #11161] Proc/Method#rcurry working like curry but in reverse order
    - 3-year-old proposal with absolutely no reaction. I've added real-life examples there several months ago.
- [Feature #14390] UnboundMethod#to_proc
- [Feature #14423] Enumerator from single object (Object#enumerate)

Misc:

- [Bug #14575] Switch Range#=== to use cover? instead of include?

Docs (probably not to discuss on development meeting, but I am not sure where I can post it to have it more visible)

- Method: https://bugs.ruby-lang.org/issues/14483
- Proc: https://bugs.ruby-lang.org/issues/14610
- MatchData: https://bugs.ruby-lang.org/issues/14450
- yield_self: https://bugs.ruby-lang.org/issues/14436

#5 - 04/19/2018 03:42 PM - mame (Yusuke Endoh)

https://bugs.ruby-lang.org/issues/14699 Subtle behaviors with endless range

#6 - 04/19/2018 04:19 PM - shevegen (Robert A. Heiler)

Just one comment:

>> From this time, we use a ticket to make dev-meeting agenda page

\o/

I think this is a good change. However had, I also think that the
wiki is very good for an "organized back-log", together with the
log summary. Not everyone can watch live edits, if only due to
time-zone issues :D but I am sure many people are interested which
changes get made to ruby, approved, rejected, request for more
comments etc... so I hope that the wiki can still be kept for
such organizational purposes, even if the tickets supercede
(part of) the wiki in this regards.

#7 - 04/19/2018 06:01 PM - greggzst (Grzegorz Jakubiak)

https://bugs.ruby-lang.org/issues/14473 Add Range#subrange?

#8 - 04/20/2018 10:11 AM - naruse (Yui NARUSE)

shevegen (Robert A. Heiler) wrote:

> Just one comment:
>
>> From this time, we use a ticket to make dev-meeting agenda page
>
> I think this is a good change. However had, I also think that the
> wiki is very good for an "organized back-log", together with the
> log summary. Not everyone can watch live edits, if only due to
> time-zone issues :D but I am sure many people are interested which
> changes get made to ruby, approved, rejected, request for more
> comments etc... so I hope that the wiki can still be kept for
> such organizational purposes, even if the tickets supercede
> (part of) the wiki in this regards.

This changed is just applied for the agenda page like https://bugs.ruby-lang.org/projects/ruby/wiki/DevelopersMeeting20180419Japan.

The wiki has an issue, other than committers cannot edit.
It is because spammers sometimes destroyed the wiki pages.
To prevent such attacks, I limited the privilege of editing wiki.

Now we use issues as agenda pages.
People can comment the issue to request adding a topic.
Committers can edit the description of issue like wiki.
(in other words we use issue's description like wiki)
This is what the decision intended.

#9 - 04/23/2018 07:02 AM - baweaver (Brandon Weaver)

zverok (Victor Shepelev) wrote:

>> your favorite ticket numbers you want to ask to discuss with your SHORT comment or summary.
>
> Functional programming:
>
> - https://bugs.ruby-lang.org/issues/6284 - add composition for Proc, long discussed and not acted upon
> - https://bugs.ruby-lang.org/issues/13581 - syntax sugar for method reference, likewise

- Pair of related tickets: https://bugs.ruby-lang.org/issues/14390 (UnboundMethod#to_proc) + https://bugs.ruby-lang.org/issues/11161 (rcurry), allowing more declarative "blocks with params" usage without any new syntax.
- https://bugs.ruby-lang.org/issues/14423 -- Object#enumerate producer of infinite sequences.

Misc:

- https://bugs.ruby-lang.org/issues/14575 - switch Range#=== to use cover?

Docs (probably not to discuss on development meeting, but I am not sure where I can post it to have it more visible)

- Method: https://bugs.ruby-lang.org/issues/14483
- Proc: https://bugs.ruby-lang.org/issues/14610
- MatchData: https://bugs.ruby-lang.org/issues/14450
- yield_self: https://bugs.ruby-lang.org/issues/14436

I would second this list, especially #6284 (composition of Procs) and #13581 (syntax sugar for method reference)

#10 - 04/25/2018 01:47 AM - baweaver (Brandon Weaver)

I would also like to propose that this be discussed as it recently came up: https://bugs.ruby-lang.org/issues/14709

Though it may be premature.

#11 - 04/25/2018 01:54 AM - ko1 (Koichi Sasada)

Hi,

I will copy & paste topics to ticket's "description" later so that please follow the format to make easy to copy&paste.

mrkn's exmaple is perfect:

```
* [Feature #14697] Introducing Range#% as an alias to Range#step (mrkn)
  * Matz already commented LGTM. Please judge whether this is acceptable.
```

Thank you for your help.

#12 - 04/25/2018 08:03 AM - zverok (Victor Shepelev)

> please follow the format to make easy to copy&paste.

ko1 (Koichi Sasada) I've updated my comment with a list of "functional programming" tickets.

#13 - 04/30/2018 10:48 AM - mrkn (Kenta Murata)

- Description updated

#14 - 04/30/2018 10:49 AM - mrkn (Kenta Murata)

- Description updated

#15 - 05/01/2018 10:29 AM - duerst (Martin Dürst)

- [Feature #14724] chains of inequalities
  - Proposal by gotoken (Kentaro Goto) to allow 0 <= a < 10 as a shortcut of 0<=a && a<10, and so on; patch by Nobu avaliable

#16 - 05/11/2018 10:02 AM - ana06 (Ana Maria Martinez Gomez)

Hi!

I would that the following issues are discussed in the meeting:

- [Feature #14097] Add union and difference to Array (ana06)
  - Addition of two new methods in aim of readability, ease of use, efficiency and consistence. There are already PR for both methods.
- [Feature #14105] Introduce xor as alias for Set#^ (ana06)
  - The exclusive or is known as xor in mathematics, hardware and many other programming languages. An alias in Set (and maybe other classes) in aim of readability and ease of use.

I would also really like to join the meeting if it is possible ;)

#17 - 05/15/2018 10:12 AM - normalperson (Eric Wong)

Please discuss what cross-thread IO#close semantics should be
for threads stuck in IO.select and IO.copy_stream:
https://bugs.ruby-lang.org/issues/14760

Thank you. I am ambivalent about the outcome, but it would
help me plan support for future features like auto-fiber.

#18 - 05/16/2018 07:58 AM - ko1 (Koichi Sasada)

- Description updated

#19 - 05/16/2018 08:01 AM - ko1 (Koichi Sasada)

Hi all,

The format is:

```
* [Ticket info] Ticket-Title (proposer)
  * Short-comment
```

And I can copy&paste :)
I edited with some editions.

Thank you for your cooperation.
Koichi

#20 - 05/16/2018 09:31 AM - ana06 (Ana Maria Martinez Gomez)

ko1 (Koichi Sasada) I edited my comment, I hope it is fine now. ;)

BTW, is it possible I join the meeting? It would be fine if I join only for those two issues. :)

#21 - 05/16/2018 01:50 PM - shyouhei (Shyouhei Urabe)

Hello, this is my private feeling so please don't take this as a canon.

I think it's a good idea to make the meeting more open. To do so several technical issues are there to be fixed. First, there is no extra staff to
welcome you right now -- like we don't have crew to livecast our meeting. You might say it's okay to be "best effort". But the developers at the
meeting are there to discuss the issues. They are not there to make "effort" on that point. Second, I think there are chances for us to welcome only
one person but very skeptic if it works for, say, 20 people. The meeting obviously does not scale. What to do then? We have to select someone by
something? That doesn't sound well.

That being said, I think it's a good idea to make the meeting more open. The problems I wrote above exist but they are only being technical I believe.
I hope we can reroute them... Maybe we can't make them for tomorrow meeting though.

#22 - 05/16/2018 03:04 PM - ana06 (Ana Maria Martinez Gomez)

shyouhei (Shyouhei Urabe)

> I think there are chances for us to welcome only one person but very skeptic if it works for, say, 20 people. The meeting obviously does not
> scale. What to do then? We have to select someone by something? That doesn't sound well.

Well, if nobody else asked until now to join the meeting, maybe nobody also asks it in future. And it is not a problem that exist at the moment. ;)

Anyway, if it is not possible that I join, I completely understand it. I am already happy that there is a discussion about those issues. :) And if it is
possible, jut let me now how! ;)

> I think it's a good idea to make the meeting more open.

+1

#23 - 05/16/2018 05:18 PM - naruse (Yui NARUSE)

ana06 (Ana Maria Martinez Gomez) wrote:

> shyouhei (Shyouhei Urabe)
>
> > I think there are chances for us to welcome only one person but very skeptic if it works for, say, 20 people. The meeting obviously does not
> > scale. What to do then? We have to select someone by something? That doesn't sound well.
>
> Well, if nobody else asked until now to join the meeting, maybe nobody also asks it in future. And it is not a problem that exist at the moment. ;)

Anyway, if it is not possible that I join, I completely understand it. I am already happy that there is a discussion about those issues. :) And if it is possible, jut let me now how! ;)

Thank you for your passion and understanding.
At this time this meeting is closed because of both technical and human resource problems.
Technical issues are as shyouhei said.

Human resource issues are because this meeting is held based on voluntary.
Because of this, the meeting includes some internal topics for example security, money for ruby related sites and machines, release engineering and so on.
Ideally such topics should be in separated meeting but we don't have enough resource.
(note that more than half of the regular member are still non full time Ruby comitter)
To keep this meeting sustainable, it is still closed for a while.
(the experiment to use redmine issues to handle topics of this meeting is also intended to make this meeting open without increasing the cost)

Of course I wish this meeting can be public in the future...

#24 - 05/17/2018 07:22 AM - nobu (Nobuyoshi Nakada)

- Description updated

#25 - 05/17/2018 08:16 AM - nobu (Nobuyoshi Nakada)

- Description updated

#26 - 05/17/2018 08:52 AM - nobu (Nobuyoshi Nakada)

- Description updated

#27 - 05/17/2018 12:11 PM - hsbt (Hiroshi SHIBATA)

- Description updated

#28 - 05/17/2018 12:30 PM - naruse (Yui NARUSE)

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

#29 - 05/17/2018 12:31 PM - naruse (Yui NARUSE)

- Status changed from Open to Closed

#30 - 05/18/2018 01:27 AM - shyouhei (Shyouhei Urabe)

And here you are the log
https://docs.google.com/document/d/e/2PACX-1vR2LdBE87iEcEsVuUUr0G2L6LxSPeGMg_0oeHeh0HYmX36iIa9zkWYlFHilH5D4I_RBJpQnr09yOZaE/pub

#31 - 05/18/2018 02:25 AM - ko1 (Koichi Sasada)

- Description updated

#32 - 05/18/2018 10:26 PM - shevegen (Robert A. Heiler)

Thanks for the link to the log shyouhei!