The next dev meeting

Date: 2022/04/21 13:00-17:00 (JST)

- 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 record of the discussion in the 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 of the meeting 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.
```

- It is recommended to add a comment by 2022/04/18. 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

History

#1 - 03/22/2022 05:35 AM - mame (Yusuke Endoh)
- Related to Misc #14770: [META] DevelopersMeeting added

#2 - 03/24/2022 09:23 PM - kddeisz (Kevin Newton)
- [Feature #18654] Enhancements to prettyprint (kddnewton)
  - PrettyPrint currently immediately pushes content onto the output buffer and then inserts newlines where necessary. This makes it impossible to change the output based on how groups are broken. This commit changes the behavior to first build up the entire doc node tree, and the lay it out once it knows all of the content. This allows a lot more flexibility and new nodes like ItBreak, BreakParent, and Trim. This flexibility is necessary to support formatting Ruby itself as in https://github.com/ruby-syntax-tree/syntax_tree.

#3 - 03/24/2022 09:25 PM - kddeisz (Kevin Newton)
- [Feature #18642] Named ripper fields (kddnewton)
  - This commit adds a new subclass for ripper that functions similarly to SexpBuilderPP but uses classes with named fields instead of arrays.
It additionally includes location information for all of the nodes, and implements pretty printing and pattern matching. This makes it much easier to work with Ripper, as you can get handed an AST with full documentation. (This addresses a lot of the complaints raised in #14844.)

#4 - 03/30/2022 12:12 PM - Eregon (Benoit Daloze)
- [Feature #18668] Merge io-nonblock gems into core (erogon)
  - Let's do it?
  - Unlike io-wait these methods seem all stable and well established.

#5 - 04/11/2022 05:01 PM - jeremyevans0 (Jeremy Evans)
- [Bug #18628] Link contributing is broken (jeremyevans0)
  - Also affects #18665. Problem is that ruby uses the --page-dir rdoc option.
  - Any documentation generation that does not use the --page-dir rdoc option has broken links.
  - I propose to remove the --page-dir rdoc option, and fix any internal documentation that relies on it.
  - This will allow externally generated documentation (which is unlikely to use --page-dir) to have working links.
- [Bug #18153] (nil..nil).cover?(x) is true for all x since beginless ranges were introduced (jeremyevans0)
  - How should cover? work for ranges unbounded in both directions?
  - In 2.7, with the introduction of beginless ranges, it started always returning true, instead of returning false.
  - I propose to reject this bug, and keep the current behavior.
- [Bug #18629] block args array splatting assigns to higher scope _ var (jeremyevans0)
  - I believe this is expected behavior. Can we confirm that and reject this bug?
- [Bug #18665] const_get still looks in Object, while lexical constant lookup no longer does (jeremyevans0)
  - Can we decide whether const_get should operate like a relative lookup or an absolute lookup?
  - Currently, it operates as a relative lookup, so Array.const_get(:String) is like class Array; String; end and works.
  - If switched to an absolute lookup, Array.const_get(:String) would be like Array::String, and would raise an exception.
- [Bug #18649] Enumerable#first breaks out of the incorrect block when across threads (jeremyevans0)
  - Can we specify how Enumerator#first should work across threads?
  - It currently has definitely incorrect behavior, jumping out of block it should not jump out of.
  - I have a patch that fixes the definitely incorrect behavior, but still passes a value to the calling thread, which may be undesireable.

#6 - 04/13/2022 07:50 AM - byroot (Jean Boussier)
- [Feature #18683] Allow to create hashes with a specific capacity.
  - Useful for various parsers and deserializers that know the Hash size in advance, such as Redis protocol, msgpack, etc.
  - When creating large hashes of known size, lots of needless reallocations are wasteful.
  - Ruby interface: Hash.new(default = nil, capacity = nil)
  - C-Ext interface: rb_hash_new_capa(long)

#7 - 04/19/2022 04:36 AM - knu (Akinori MUSHA)
- [Feature #18685] Enumerator.product: Cartesian product of enumerables

#8 - 04/19/2022 07:00 PM - Eregon (Benoit Daloze)
- [Bug #18729] Method#owner and UnboundMethod#owner are incorrect after using Module#public/protected/private (erogon)
  - OK to fix this?
  - Any insight why the behavior was inconsistent for private/protected/public as that's the only case where owner is not "the module on which the method is defined/part of its method table"?
  - Could someone review the PR?

#9 - 04/22/2022 01:23 AM - mame (Yusuke Endoh)
- Description updated
- Status changed from Open to Closed