Ruby master - Feature #14468
Add Proc#dig
02/12/2018 09:04 PM - bradleybuda (Bradley Buda)

<table>
<thead>
<tr>
<th>Status:</th>
<th>Rejected</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority:</td>
<td>Normal</td>
</tr>
<tr>
<td>Assignee:</td>
<td>matz (Yukihiro Matsumoto)</td>
</tr>
<tr>
<td>Target version:</td>
<td></td>
</tr>
</tbody>
</table>

**Description**

Since Proc already responds to [], it would be cool if Procs could participate in a recursive dig. Like this:

**Current Behavior:**

```ruby
obj = [
  0,
  {
    a: ->(x) { x * 2 },
    b: "c"
  },
]
```

```ruby
obj[1][:a][4] == 8 # true
obj.dig(1, :a, 4) == 8 # TypeError (Proc does not have #dig method)
```

**Desired behavior:**

```ruby
obj.dig(1, :a, 4) == 8 # true
```

I am willing to implement this but I wanted to see if the devs think it is a good idea first. If there are no objections, I'll put together a patch.

**History**

#1 - 02/13/2018 12:24 AM - shevegen (Robert A. Heller)

I am not one of the devs so this is just my own opinion.

I sort of agree with the proposal; not so much because I would need it (my style is very awkward and I do not use procs that often; I used them for callbacks in the past in ruby-gtk applications but other than that, I myself rarely use procs and lambdas).

In the worst case perhaps the proposal here can be discussed (and thus added) it can be added to the upcoming ruby developer meeting, to get what matz thinks. In the end you only have to convince matz, primarily. :)

The reason why I think it makes sense is because .dig was used to "dig into the data structure", and Procs are objects too, so why not.

Hash#dig:

http://ruby-doc.org/core-2.3.0_preview1/Hash.html#method-i-dig

I think for symmetry it would make sense, which is why I agree.

(On a side note, it seems as if everyone is busy with mjlt; I noticed that in the last some days where mjlt seems to be the main focus here too. Wonder if I am the only one with that impression; can't wait to have mjlt merged in completely. :) )

Ruby developer meeting may be at:

https://bugs.ruby-lang.org/projects/ruby/wiki/DevelopersMeeting20180220Japan

Perhaps if you think it should be added, someone can add it. (I don't want to suggest it because it is your issue, not
mine, and I can not decide for you - I only wanted to mention it, since it may be easier to get it to discuss than perhaps wait for a bit longer afterwards, depending on how busy the ruby core team is, e. g. see the mjlt work right now.)

#2 - 02/13/2018 04:32 AM - marcandre (Marc-Andre Lafortune)
- Assignee set to matz (Yukihiro Matsumoto)

I don't oppose it, but I'm not very enthusiastic about it. Integer and String also respond_to [], and I don't quite see dig for these either.

#3 - 02/13/2018 10:24 AM - sawa (Tsuyoshi Sawada)

I am also not a fan of this particular case, but I think it makes sense to let dig call fetch (or [[]]) recursively, whenever the element responds to it.

#4 - 02/20/2018 06:09 AM - matz (Yukihiro Matsumoto)
- Status changed from Open to Rejected

The original intention for dig is a short-hand traversal of array-hash trees. I don't think it's normal to use proc objects as tree nodes.

Matz.