

Ruby master - Bug #9242

Rdoc detection of aliases

12/12/2013 04:46 AM - marcandre (Marc-Andre Lafortune)

Status: Closed	
Priority: Normal	
Assignee: drbrain (Eric Hodel)	
Target version:	
ruby -v: 2.1.0 preview	Backport:
Description	
The online doc appears to be making bad detection of aliases.	
For example, Array#map and Array#collect are not marked as aliases (on either ruby-doc.org or docs.ruby-lang.org)	
On the other hand, when aliases are detected, the generated method interface and the name of the method it is aliases to are always the same for both methods.	
For example, the doc for Array#find_index says it is an alias to find_index and has interface showing index (again on both sites), e.g. http://docs.ruby-lang.org/en/2.0.0/Array.html#method-i-find_index	
It might also be a good idea to always have doc examples using both forms in case of aliases.	

History

#1 - 12/12/2013 06:10 AM - Hanmac (Hans Mackowiak)

Array#map and Array#collect are not real aliases
both methods are defined with the exact same method body function but both are not calling each other,
so changing one method does not break the other

#2 - 12/12/2013 06:53 AM - zzak (Zachary Scott)

- Status changed from Open to Assigned
- Assignee changed from drbrain (Eric Hodel) to zzak (Zachary Scott)

Alias detection was updated in the pre-release of RDoc 4.1.0, can you try to test it?

If it's these methods are using the same method, we can easily fix this by adding another signature to the call-seq example. What do you think?

#3 - 12/12/2013 01:09 PM - drbrain (Eric Hodel)

- Assignee changed from zzak (Zachary Scott) to drbrain (Eric Hodel)
- Target version set to 2.6

As Hanmac noted, they aren't real aliases:

```
$ grep rb_ary_collect array.c
rb_ary_collect(VALUE ary)
rb_ary_collect_bang(VALUE ary)
rb_define_method(rb_cArray, "collect", rb_ary_collect, 0);
rb_define_method(rb_cArray, "collect!", rb_ary_collect_bang, 0);
rb_define_method(rb_cArray, "map", rb_ary_collect, 0);
rb_define_method(rb_cArray, "map!", rb_ary_collect_bang, 0);
```

But I guess this is for performance reasons (can someone confirm?). I suppose RDoc should treat these as aliases.

#4 - 12/12/2013 01:19 PM - marcandre (Marc-Andre Lafortune)

There really shouldn't be a difference between using rb_define_alias and using the same rb_define_method.

In Ruby, Array.instance_method(:map) == Array.instance_method(:collect) returns true (even when not defined using rb_define_alias), and they should be treated as aliases.

Moreover, the problem isn't there, since Array#find_index and index are not defined using rb_define_alias either but the generated doc sees them as aliases.

#5 - 02/19/2014 01:06 AM - zzak (Zachary Scott)

- *Category changed from doc to lib*

#6 - 12/25/2017 06:15 PM - naruse (Yui NARUSE)

- *Target version deleted (2.6)*

#7 - 09/11/2019 12:03 AM - jeremyevans0 (Jeremy Evans)

- *Backport deleted (1.9.3: UNKNOWN, 2.0.0: UNKNOWN)*

This is still an issue in the master branch. I've added a pull request for rdoc that will treat `rb_define_method` to the same C function as an alias (similar to `rb_define_alias`): <https://github.com/ruby/rdoc/pull/742>

#8 - 03/06/2020 09:31 PM - jeremyevans0 (Jeremy Evans)

- *Status changed from Assigned to Closed*