

Ruby master - Feature #14332

Module.used_refinements to list refinement modules

01/07/2018 11:31 PM - Eregon (Benoit Daloz)

Status:	Assigned
Priority:	Normal
Assignee:	shugo (Shugo Maeda)
Target version:	
Description	
<p>Module.used_modules was added in #7418. But I think Module.used_refinements is more useful or at least complementary.</p> <p>Refinements were implemented in TruffleRuby, and I found Module.used_refinements so useful that I left it there.</p> <p>Instead of listing namespace modules (arguments to #using), it lists the refinement modules:</p> <pre>module Json refine Integer do def to_json to_s end end end refine String do def to_json inspect end end module Fact refine Integer do def fact self <= 1 ? 1 : self * (self-1).fact end end end using Json p Module.used_modules # => [Json] p Module.used_refinements # => [#<refinement:Integer@Json>, #<refinement:String@Json>] using Fact p Module.used_modules # => [Json, Fact] p Module.used_refinements # => [#<refinement:Integer@Fact>, #<refinement:Integer@Json>, #<refinement:String@Json>]</pre> <p>This shows which classes are refined and by which refinement namespace. It also shows if a class has multiple active refinements in the scope. And, last but not least, the name of the method is contains "refinements". I find used_modules hard to remember and it doesn't sound related to refinements from the name (while looking at the Module's methods).</p>	

History

#1 - 02/19/2018 10:02 AM - shugo (Shugo Maeda)

- Status changed from Open to Assigned

Eregon (Benoit Daloz) wrote:

Module.used_modules was added in [#7418](#).
But I think Module.used_refinements is more useful or at least complementary.

I'm for it, but I'd like to hear others' opinions.

#2 - 12/10/2018 06:59 AM - naruse (Yui NARUSE)

- *Target version deleted (2.6)*