

Ruby trunk - Bug #8930

some benchmark programs are slow since r42822

09/20/2013 07:04 PM - authorNari (Narihiro Nakamura)

Status: Closed	
Priority: Normal	
Assignee: charliesome (Charlie Somerville)	
Target version: 2.1.0	
ruby -v: ruby 2.1.0dev (2013-09-04 trunk r42822) [x86_64-linux]	Backport: 1.9.3: UNKNOWN, 2.0.0: UNKNOWN
Description	
Hi.	
I find some benchmark programs are slow since r42822 .	
<pre>% time ./miniruby -v benchmark/bm_vm3_clearmethodcache.rb ruby 2.1.0dev (2013-09-04 trunk 42910) [x86_64-linux] ./miniruby -v benchmark/bm_vm3_clearmethodcache.rb 0.42s user 0.00s system 98% cpu 0.451 total</pre>	
<pre>% time ./miniruby -v benchmark/bm_vm3_clearmethodcache.rb ruby 2.1.0dev (2013-09-04 trunk r42822) [x86_64-linux] ./miniruby -v benchmark/bm_vm3_clearmethodcache.rb 0.48s user 0.01s system 98% cpu 0.495 total</pre>	
In addition to the above:	
<ul style="list-style-type: none">• vm2_poly_method• vm2_send• so_object• etc... (If you want to know all slow benchmark programs, please check this site http://www.narihiro.info/crubybenci/)	
If this isn't intended, could you check it, charliesome or ko1?	
Thanks	

Associated revisions

Revision f0888281 - 09/24/2013 05:02 AM - charliesome (Charlie Somerville)

- class.c (class_alloc): remove mc_tbl
- gc.c (obj_free): ditto
- internal.h (struct rb_classex_struct): ditto
- method.h (rb_method_entry): remove ent param
- vm_method.c: restore the global method cache. Per class cache tables turned out to be far too slow.

[ruby-core:57289] [Bug #8930]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@43027 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 43027 - 09/24/2013 05:02 AM - charliesome (Charlie Somerville)

- class.c (class_alloc): remove mc_tbl
- gc.c (obj_free): ditto
- internal.h (struct rb_clasext_struct): ditto
- method.h (rb_method_entry): remove ent param
- vm_method.c: restore the global method cache. Per class cache tables turned out to be far too slow.

[ruby-core:57289] [Bug #8930]

Revision 43027 - 09/24/2013 05:02 AM - charliesome (Charlie Somerville)

- class.c (class_alloc): remove mc_tbl
- gc.c (obj_free): ditto
- internal.h (struct rb_clasext_struct): ditto
- method.h (rb_method_entry): remove ent param
- vm_method.c: restore the global method cache. Per class cache tables turned out to be far too slow.

[ruby-core:57289] [Bug #8930]

Revision 43027 - 09/24/2013 05:02 AM - charliesome (Charlie Somerville)

- class.c (class_alloc): remove mc_tbl

- gc.c (obj_free): ditto
- internal.h (struct rb_classex_struct): ditto
- method.h (rb_method_entry): remove ent param
- vm_method.c: restore the global method cache. Per class cache tables turned out to be far too slow.

[ruby-core:57289] [Bug #8930]

Revision 43027 - 09/24/2013 05:02 AM - charliesome (Charlie Somerville)

- class.c (class_alloc): remove mc_tbl
- gc.c (obj_free): ditto
- internal.h (struct rb_classex_struct): ditto
- method.h (rb_method_entry): remove ent param
- vm_method.c: restore the global method cache. Per class cache tables turned out to be far too slow.

[ruby-core:57289] [Bug #8930]

Revision 43027 - 09/24/2013 05:02 AM - charliesome (Charlie Somerville)

- class.c (class_alloc): remove mc_tbl
- gc.c (obj_free): ditto
- internal.h (struct rb_classex_struct): ditto
- method.h (rb_method_entry): remove ent param
- vm_method.c: restore the global method cache. Per class cache tables turned out to be far too slow.

[ruby-core:57289] [Bug #8930]

History

#1 - 09/24/2013 02:02 PM - charliesome (Charlie Somerville)

- Status changed from Open to Closed

- % Done changed from 0 to 100

This issue was solved with changeset [r43027](#).

Narihiro, thank you for reporting this issue.

Your contribution to Ruby is greatly appreciated.

May Ruby be with you.

- class.c (class_alloc): remove mc_tbl
- gc.c (obj_free): ditto
- internal.h (struct rb_clasext_struct): ditto
- method.h (rb_method_entry): remove ent param
- vm_method.c: restore the global method cache. Per class cache tables turned out to be far too slow.

[Bug [#8930](#)]