Ruby master - Bug #7635

debug_inspector API segfaults when opened from inside an eval frame

12/29/2012 10:52 PM - charliesome (Charlie Somerville)

<table>
<thead>
<tr>
<th>Status:</th>
<th>Closed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority:</td>
<td>Normal</td>
</tr>
<tr>
<td>Assignee:</td>
<td>ko1 (Koichi Sasada)</td>
</tr>
<tr>
<td>Target version:</td>
<td>2.0.0</td>
</tr>
<tr>
<td>ruby -v:</td>
<td>ruby 2.0.0dev (2012-12-29 trunk 38656) [x86_64-darwin11.4.0]</td>
</tr>
<tr>
<td>Backport:</td>
<td></td>
</tr>
</tbody>
</table>

### Description

=begin

(({rb_debug_inspector_open})) segfaults when it is called from inside an eval frame.

Using this C extension:

#include "ruby/ruby.h"

static VALUE cb()
{
  return Qnil;
}

static VALUE debug_inspector()
{
  return rb_debug_inspector_open(cb, NULL);
}

void Init_debug_inspector()
{
  rb_define_global_function("debug_inspector", debug_inspector, 0);
}

Calling (({debug_inspector})) from the top level and from within methods works fine, calling inside eval from the top level works fine, but calling inside eval from within a method segfaults.

Here is a test program that segfaults:

```ruby
require './debug_inspector' # the c extension above

def x
  eval "debug_inspector"
end

x
=end
```

### Associated revisions

Revision 18e01f63 - 01/29/2013 08:25 AM - ko1 (Koichi Sasada)

- vm_backtrace.c: fix issue of rb_debug_inspector_open(). The order of making binding should be stack (frame) top to bottom. [Bug #7635] And also fix issue of collecting klass. Collecting klass is same as TracePoint#defined_class. (previous version, it returns T_ICLASS (internal objects).
- test/-ext-/debug/test_debug.rb: add a test.
- ext/-test-/debug/extconf.rb, init.c, inspector.c: ditto.
- vm_backtrace.c: remove magic number and add enum CALLER_BINDING.*.
- vm_backtrace.c, include/ruby/debug.h: add new C api (experimental) rb_debug_inspector_frame_self_get().
- vm.c, vm_core.h, vm_trace.c: move decl. of rb_vm_control_frame_id_and_class() and constify first parameter.

09/16/2021
vm_backtrace.c: fix issue of rb_debug_inspector_open(). The order of making binding should be stack (frame) top to bottom. [Bug #7635] And also fix issue of collecting klass. Collecting klass is same as TracePoint#defined_class. (previous version, it returns T_ICLASS (internal objects).

- test/ext-debug/test_debu.rb: add a test.
- ext-test/debug/extconf.rb, init.c, inspector.c: ditto.
- vm_backtrace.c: remove magic number and add enum CALLER_BINDING_*.
- vm_backtrace.c, include/ruby/debug.h: add new C api (experimental) rb_debug_inspector_frame_self_get().
- vm.c, vm_core.h, vm_trace.c: move decl. of rb_vm_control_frame_id_and_class() and constify first parameter.

vm_backtrace.c: remove magic number and add enum CALLER_BINDING_*.
vm.c, vm_core.h, vm_trace.c: move decl. of rb_vm_control_frame_id_and_class() and constify first parameter.

vm_backtrace.c: remove magic number and add enum CALLER_BINDING_*.
vm.c, vm_core.h, vm_trace.c: move decl. of rb_vm_control_frame_id_and_class() and constify first parameter.

vm_backtrace.c: remove magic number and add enum CALLER_BINDING_*.
vm.c, vm_core.h, vm_trace.c: move decl. of rb_vm_control_frame_id_and_class() and constify first parameter.
Ah.. I missed this ticket.

#2 - 01/29/2013 05:25 PM - ko1 (Koichi Sasada)
- Status changed from Assigned to Closed
- % Done changed from 0 to 100

This issue was solved with changeset r38970.
Charlie, thank you for reporting this issue.
Your contribution to Ruby is greatly appreciated.
May Ruby be with you.

- vm_backtrace.c: fix issue of rb_debug_inspector_open(). The order of making binding should be stack (frame) top to bottom. [Bug #7635] And also fix issue of collecting klass. Collecting klass is same as TracePoint#defined_class. (previous version, it returns T_ICLASS (internal objects).
- test/-ext-/debug/test_debug.rb: add a test
- ext/-test-/debug/extconf.rb, init.c, inspector.c: ditto.
- vm_backtrace.c: remove magic number and add enum CALLER_BINDING_*.
- vm_backtrace.c, include/ruby/debug.h: add new C api (experimental) rb_debug_inspector_frame_self_get().
- vm.c, vm_core.h, vm_trace.c: move decl. of rb_vm_control_frame_id_and_class() and constify first parameter.