Description

The patch (committed) for Issue #4614 made this change possible.

Lowering stack size allows users on memory-constrained 32-bit machines to spawn more native threads (which are easier (IMHO) to use than Fibers).

Setting RUBY_STACK_MIN_LIMIT to 48K seems to work even with check, test-rubyspec, and benchmark-each targets. However, I'm keeping it at 64K since that is what Symbian uses, so maybe there's some code paths that need 64K.

I started experiencing failures with the Ruby tests with 44K on x86_64, so 44K is definitely not safe. Much more effort would be required to make Ruby work with smaller stacks than 48K.

Also pullable via git: git pull git://bogomips.org/ruby.git stack-reduce

History

#1 - 06/10/2011 10:55 PM - ko1 (Koichi Sasada)
- Assignee set to ko1 (Koichi Sasada)

GC eats large stack size if there are nested object (deep nested array and so on). I'm not sure but I think this is why we allocate 512KB for stack.

Any comments?

I think that Thread.new should accept stack size.

#2 - 06/11/2011 04:23 PM - normalperson (Eric Wong)

Koichi Sasada redmine@ruby-lang.org wrote:

GC eats large stack size if there are nested object (deep nested array and so on). I'm not sure but I think this is why we allocate 512KB for stack.

Good point. I didn't think of that (rare case for me). We would need a non-recursive implementation of gc_mark_children.

A simpler idea would be a dedicated marking thread with a deeper stack; but I think that's ugly.

Any comments?

I think that Thread.new should accept stack size.

I'm not sure what the API would be. While it would help me, I think it would be difficult to use and too low level for Ruby.

Meanwhile I can rebuild Ruby or use a trivial LD_PRELOAD:
http://yhbt.net/libministack.c

03/17/2020
Ad-hoc solution, but environment variable (such as RUBY_THREAD_MACHINE_STACK_SIZE) is enough for this issue?

Also, it seems the recursive structure problem with gc_mark_children() is gone since r37075. Probably better to discuss #6695 instead.

I think environment variable is too ugly, but it's your (or matz's) choice to support it.

Welcome to volunteer for helping him.

I add environment variable at r38479.

- RUBY_THREAD_VM_STACK_SIZE: vm stack size used at thread creation.
  default: 128KB (32bit CPU) or 256KB (64bit CPU).
- RUBY_THREAD_MACHINE_STACK_SIZE: machine stack size used at thread creation.
  default: 512KB or 1024KB.
- RUBY_FIBER_VM_STACK_SIZE: vm stack size used at fiber creation.
  default: 64KB or 128KB.
- RUBY_FIBER_MACHINE_STACK_SIZE: machine stack size used at fiber creation.
  default: 256KB or 256KB.

I want to ask feedback about this default setting.

Thanks!

Btw, you put a giant "if false" around test/ruby/test_fiber.rb with that commit. Was that intentional?
I want to ask feedback about this default setting.

I'm fine with the current defaults (including the r38592 update)

#10 - 02/13/2013 04:00 PM - ko1 (Koichi Sasada)
- Status changed from Feedback to Closed

Sorry I missed your comment.
And the 'if false' sentences may be removed.

Files

<table>
<thead>
<tr>
<th>File Name</th>
<th>Size</th>
<th>Date</th>
<th>Author</th>
</tr>
</thead>
<tbody>
<tr>
<td>0001-thread_pthread.c-lower-RUBY_STACK_MIN_LIMIT-to-64K.patch</td>
<td>34 KB</td>
<td>04/26/2011</td>
<td>normalperson (Eric Wong)</td>
</tr>
</tbody>
</table>