

Ruby trunk - Bug #9121

[PATCH] Remove rbtree implementation of SortedSet due to performance regression

11/18/2013 09:24 AM - xshay (Xavier Shay)

| | |
|--|---|
| Status: Assigned | |
| Priority: Normal | |
| Assignee: knu (Akinori MUSHA) | |
| Target version: Next Major | |
| ruby -v: 2.0.0-p247 | Backport: 1.9.3: UNKNOWN, 2.0.0: UNKNOWN |
| Description | |
| rbtree is slower than the pure ruby version. | |
| I have provided benchmarks and a patch here: https://github.com/ruby/ruby/pull/451 | |
| <pre>ruby sorted_set_benchmark.rb using rbtree user system total real #add 0.010000 0.000000 0.010000 (0.016446) #delete 0.020000 0.000000 0.020000 (0.013248) #include? 1000 items 0.010000 0.000000 0.010000 (0.011822) #include? 2000 items 0.020000 0.000000 0.020000 (0.012572) #include? 3000 items 0.020000 0.000000 0.020000 (0.013610) #include? 4000 items 0.020000 0.000000 0.020000 (0.014295) #include? 5000 items 0.010000 0.000000 0.010000 (0.018024) #to_a 1000 items 0.580000 0.020000 0.600000 (0.616104) #to_a 2000 items 1.170000 0.040000 1.210000 (1.213406) #to_a 3000 items 1.730000 0.030000 1.760000 (1.773069) #to_a 4000 items 2.370000 0.040000 2.410000 (2.420450) #to_a 5000 items 2.920000 0.050000 2.970000 (2.975497) ruby sorted_set_benchmark.rb NOT using rbtree user system total real #add 0.010000 0.000000 0.010000 (0.007889) #delete 0.010000 0.000000 0.010000 (0.004631) #include? 1000 items 0.000000 0.000000 0.000000 (0.005060) #include? 2000 items 0.010000 0.000000 0.010000 (0.005950) #include? 3000 items 0.010000 0.000000 0.010000 (0.005814) #include? 4000 items 0.010000 0.000000 0.010000 (0.005993) #include? 5000 items 0.010000 0.000000 0.010000 (0.006923) #to_a 1000 items 0.000000 0.000000 0.000000 (0.001863) #to_a 2000 items 0.000000 0.000000 0.000000 (0.002145) #to_a 3000 items 0.000000 0.000000 0.000000 (0.002129) #to_a 4000 items 0.000000 0.000000 0.000000 (0.002265) #to_a 5000 items 0.000000 0.000000 0.000000 (0.002428)</pre> | |
| Related issues: | |
| Related to Ruby trunk - Feature #2348: RBTree Should be Added to the Standard... | Rejected |

History

#1 - 11/18/2013 09:43 AM - zzak (Zachary Scott)

- Category set to lib

- Status changed from Open to Assigned

- Assignee set to knu (Akinori MUSHA)

#2 - 11/19/2013 07:04 AM - vjoel (Joel VanderWerf)

As noted at <https://github.com/ruby/ruby/pull/451#issuecomment-28741490>:

These benchmarks miss the point of using rbtree, which is to pay a small insertion cost to keep the structure sorted so that ordered lookups are fast. If you only need to perform lookups after the structure is built, then rbtree is a waste of time.

See <https://gist.github.com/vjoel/7535917>.

Also, this part of the benchmark is particularly misleading:

```
b.report("#to_a #{n} items") {
  10000.times { s.to_a }
}
```

because the non-rbtree implementation caches the #to_a output and will never drop that cache inside of the above loop.

#3 - 11/23/2013 05:51 PM - knu (Akinori MUSHA)

Thanks for your input, guys.

I think I'll drop the optional rbtree version of SortedSet for now, since rbtree is seemingly broken for the latest version of ruby.

#4 - 11/23/2013 06:09 PM - mame (Yusuke Endoh)

knu (Akinori MUSHA) wrote:

rbtree is seemingly broken for the latest version of ruby.

What do you mean? What broke rbtree?

I'm afraid it is a more serious problem than this ticket itself.

--

Yusuke Endoh mame@tsg.ne.jp

#5 - 11/23/2013 09:23 PM - zzak (Zachary Scott)

See [#7698](#) and <https://github.com/seki/Drip/issues/4>

On Nov 23, 2013, at 6:09 PM, "mame (Yusuke Endoh)" mame@tsg.ne.jp wrote:

Issue [#9121](#) has been updated by mame (Yusuke Endoh).

knu (Akinori MUSHA) wrote:

rbtree is seemingly broken for the latest version of ruby.

What do you mean? What broke rbtree?

I'm afraid it is a more serious problem than this ticket itself.

--

Yusuke Endoh mame@tsg.ne.jp

Bug [#9121](#): [PATCH] Remove rbtree implementation of SortedSet due to performance regression
<https://bugs.ruby-lang.org/issues/9121#change-43101>

Author: xshay (Xavier Shay)

Status: Assigned

Priority: Normal

Assignee: knu (Akinori MUSHA)

Category: lib

Target version:

ruby -v: 2.0.0-p247

Backport: 1.9.3: UNKNOWN, 2.0.0: UNKNOWN

rbtree is slower than the pure ruby version.

I have provided benchmarks and a patch here:

<https://github.com/ruby/ruby/pull/451>

```
ruby sorted_set_benchmark.rb
using rbtree
user system total real
#add 0.010000 0.000000 0.010000 ( 0.016446)
#delete 0.020000 0.000000 0.020000 ( 0.013248)
#include? 1000 items 0.010000 0.000000 0.010000 ( 0.011822)
```

```
#include? 2000 items 0.020000 0.000000 0.020000 ( 0.012572)
#include? 3000 items 0.020000 0.000000 0.020000 ( 0.013610)
#include? 4000 items 0.020000 0.000000 0.020000 ( 0.014295)
#include? 5000 items 0.010000 0.000000 0.010000 ( 0.018024)
#to_a 1000 items 0.580000 0.020000 0.600000 ( 0.616104)
#to_a 2000 items 1.170000 0.040000 1.210000 ( 1.213406)
#to_a 3000 items 1.730000 0.030000 1.760000 ( 1.773069)
#to_a 4000 items 2.370000 0.040000 2.410000 ( 2.420450)
#to_a 5000 items 2.920000 0.050000 2.970000 ( 2.975497)
ruby sorted_set_benchmark.rb
NOT using rbtree
user system total real
#add 0.010000 0.000000 0.010000 ( 0.007889)
#delete 0.010000 0.000000 0.010000 ( 0.004631)
#include? 1000 items 0.000000 0.000000 0.000000 ( 0.005060)
#include? 2000 items 0.010000 0.000000 0.010000 ( 0.005950)
#include? 3000 items 0.010000 0.000000 0.010000 ( 0.005814)
#include? 4000 items 0.010000 0.000000 0.010000 ( 0.005993)
#include? 5000 items 0.010000 0.000000 0.010000 ( 0.006923)
#to_a 1000 items 0.000000 0.000000 0.000000 ( 0.001863)
#to_a 2000 items 0.000000 0.000000 0.000000 ( 0.002145)
#to_a 3000 items 0.000000 0.000000 0.000000 ( 0.002129)
#to_a 4000 items 0.000000 0.000000 0.000000 ( 0.002265)
#to_a 5000 items 0.000000 0.000000 0.000000 ( 0.002428)
```

--
<http://bugs.ruby-lang.org/>

#6 - 11/23/2013 09:53 PM - knu (Akinori MUSHA)

mame (Yusuke Endoh) wrote:

knu (Akinori MUSHA) wrote:

rbtree is seemingly broken for the latest version of ruby.

What do you mean? What broke rbtree?

Try it yourself and you'll see. It relies on the internal data structure of RHash at some point of Ruby that lasted until 1.9.3.

I'm afraid it is a more serious problem than this ticket itself.

It is obviously a third-party issue and the point of this issue is that we should not depend on it any more, so I'll go ahead anyway.

#7 - 11/23/2013 10:01 PM - knu (Akinori MUSHA)

I wrote:

... and the point of this issue is that we should not depend on it any more, so I'll go ahead anyway.

Oops, I mistook this issue for something else, I just thought that's another reason to ditch the rbtree dependency albeit optional.

#8 - 11/23/2013 10:07 PM - mame (Yusuke Endoh)

zzak (Zachary Scott) wrote:

See [#7698](#) and <https://github.com/seki/Drip/issues/4>

Thank you, but it works now for me.

```
$ ruby -v
ruby 2.0.0p353 (2013-11-22 revision 43784) [x86_64-linux]
$ gem -v
2.1.11
$ gem install rbtree
Building native extensions. This could take a while...
Successfully installed rbtree-0.4.1
```

Parsing documentation for rbtree-0.4.1
Done installing documentation for rbtree after 0 seconds
1 gem installed

knu (Akinori MUSHA) wrote:

What do you mean? What broke rbtree?

Try it yourself and you'll see. It relies on the internal data structure of RHash at some point of Ruby that lasted until 1.9.3.

I'm using rbtree on 2.0.0 everyday, but I have never encountered any problem.
Where was the issue discussed? Could you show me a pointer?

I'm afraid it is a more serious problem than this ticket itself.

It is obviously a third-party issue and the point of this issue is that we should not depend on it any more, so I'll go ahead anyway.

In principle, I agree that it is not cool for a standard library to depend
a third-party library. But in fact, (a part of) set.rb has depended on rbtree.
As Joel said, some operations will become very slow if it is removed.
Isn't it a compatibility problem?

--

Yusuke Endoh mame@tsg.ne.jp

#9 - 11/23/2013 10:44 PM - knu (Akinori MUSHA)

- Target version set to Next Major

Maybe. And I noticed the second preview (not RC though) of 2.1.0 was out, so I'll postpone any change to SortedSet to the next major.

#10 - 12/01/2013 06:09 AM - tmm1 (Aman Gupta)

The following patch fixes rbtree on ruby 2.1: <https://gist.github.com/tmm1/7609371>

I emailed it to burningdowntheopera@yahoo.co.jp 8 days ago, but no response.

#11 - 12/01/2013 12:06 PM - zzak (Zachary Scott)

Thank you tmm1!

There is also rbtree2 (<https://github.com/kitak/rbtree2>), that is also seemingly unmaintained, but perhaps they will be more willing to do a release.
Ofcourse, it would be best if we could get OZAWA-san to respond to the patch and release.

#12 - 12/26/2013 12:00 AM - knu (Akinori MUSHA)

On second thought, if a working version rbtree is available out there (at least for CRuby and hopefully for JRuby) then I would like SortedSet to be
moved from the standard library to a third-party gem.

The non-rbtree implementation of SortedSet should be seen as an ad hoc substitute, not a professional, production quality implementation to provide
the performance characteristics that would be naturally expected from the name.

#13 - 01/21/2014 05:07 PM - zzak (Zachary Scott)

I just wanted to remind everyone of the discussion that took place regarding adding RBTree to stdlib in [Feature [#2348](#)]

#14 - 08/27/2014 03:29 AM - hsbt (Hiroshi SHIBATA)

- Related to Feature [#2348](#): RBTree Should be Added to the Standard Library added