

Ruby - Bug #19969

Regression of memory usage with Ruby 3.1

10/24/2023 09:47 AM - hsbt (Hiroshi SHIBATA)

Status:	Closed	Backport: 3.0: DONTNEED, 3.1: DONE, 3.2: DONE
Priority:	Normal	
Assignee:		
Target version:		
ruby -v:		
Description		
Our company that is ANDPAD, Inc. encountered to increase memory usage after upgrading Ruby 3.2 from 3.0 on our Rails application. This increase size is about 20%. My colleague found this root cause and reproduction code: <pre>\$ ruby -v -rset -e 's1 = Set.new(10000.times); s2 = Set.new(9999.times); Array.new(10000) { s1 - s2 - [0] }; puts `ps -o rss= -p #{\$\$}`.to_i' ruby 3.0.6p216 (2023-06-29 revision bdfef1958a8) +JIT [arm64-darwin22] 248096 \$ ruby -v -rset -e 's1 = Set.new(10000.times); s2 = Set.new(9999.times); Array.new(10000) { s1 - s2 - [0] }; puts `ps -o rss= -p #{\$\$}`.to_i' ruby 3.2.2 (2023-07-05 revision 2f603bc4d7) +YJIT [arm64-darwin22] 2949280</pre> Should we revert #16996 for Ruby 3.1 or later? I'm not sure this increased memory usage is reasonable with performance improvement.		

Associated revisions

Revision 9eac9d71786a8dbec520d0541a91149f01adf8ea - 11/11/2023 09:49 AM - nobu (Nobuyoshi Nakada)

[Bug #19969] Compact st_table after deleted if possible

Revision 9eac9d71786a8dbec520d0541a91149f01adf8ea - 11/11/2023 09:49 AM - nobu (Nobuyoshi Nakada)

[Bug #19969] Compact st_table after deleted if possible

Revision 9eac9d71 - 11/11/2023 09:49 AM - nobu (Nobuyoshi Nakada)

[Bug #19969] Compact st_table after deleted if possible

Revision 1cc38d5a2f84733e1c2e42548639e2891fe61e69 - 11/19/2023 01:00 PM - nagachika (Tomoyuki Chikanaga)

merge revision(s) 9eac9d71786a8dbec520d0541a91149f01adf8ea: [Backport #19969]

```
[Bug #19969] Compact st_table after deleted if possible

---
hash.c          | 19 ++++++
st.c            | 40 ++++++
test/ruby/test_hash.rb | 9 ++++++
3 files changed, 57 insertions(+), 11 deletions(-)
```

Revision 1cc38d5a2f84733e1c2e42548639e2891fe61e69 - 11/19/2023 01:00 PM - nagachika (Tomoyuki Chikanaga)

merge revision(s) 9eac9d71786a8dbec520d0541a91149f01adf8ea: [Backport #19969]

```
[Bug #19969] Compact st_table after deleted if possible

---
hash.c          | 19 ++++++
st.c            | 40 ++++++
test/ruby/test_hash.rb | 9 ++++++
3 files changed, 57 insertions(+), 11 deletions(-)
```

Revision 1cc38d5a - 11/19/2023 01:00 PM - nagachika (Tomoyuki Chikanaga)

merge revision(s) 9eac9d71786a8dbec520d0541a91149f01adf8ea: [Backport #19969]

[Bug #19969] Compact st_table after deleted if possible

```
---
hash.c          | 19 ++++++
st.c            | 40 ++++++-----
test/ruby/test_hash.rb | 9 ++++++
3 files changed, 57 insertions(+), 11 deletions(-)
```

Revision 1cae5e7ceaca7304108fdec35d4858a9e4ff7fe0 - 11/20/2023 12:11 PM - U.Nakamura

merge revision(s) 9eac9d71786a8dbec520d0541a91149f01adf8ea: [Backport #19969]

[Bug #19969] Compact st_table after deleted if possible

```
---
hash.c          | 19 ++++++
st.c            | 40 ++++++-----
test/ruby/test_hash.rb | 9 ++++++
3 files changed, 57 insertions(+), 11 deletions(-)
```

Revision 1cae5e7ceaca7304108fdec35d4858a9e4ff7fe0 - 11/20/2023 12:11 PM - U.Nakamura

merge revision(s) 9eac9d71786a8dbec520d0541a91149f01adf8ea: [Backport #19969]

[Bug #19969] Compact st_table after deleted if possible

```
---
hash.c          | 19 ++++++
st.c            | 40 ++++++-----
test/ruby/test_hash.rb | 9 ++++++
3 files changed, 57 insertions(+), 11 deletions(-)
```

Revision 1cae5e7c - 11/20/2023 12:11 PM - U.Nakamura

merge revision(s) 9eac9d71786a8dbec520d0541a91149f01adf8ea: [Backport #19969]

[Bug #19969] Compact st_table after deleted if possible

```
---
hash.c          | 19 ++++++
st.c            | 40 ++++++-----
test/ruby/test_hash.rb | 9 ++++++
3 files changed, 57 insertions(+), 11 deletions(-)
```

History

#1 - 10/24/2023 11:02 AM - nobu (Nobuyoshi Nakada)

May <https://github.com/nobu/ruby/tree/rehash-after-delete> help it?

#2 - 10/24/2023 11:03 AM - nobu (Nobuyoshi Nakada)

- Backport changed from 3.0: UNKNOWN, 3.1: UNKNOWN, 3.2: UNKNOWN to 3.0: DONTNEED, 3.1: REQUIRED, 3.2: REQUIRED

#3 - 10/24/2023 02:59 PM - Eregon (Benoit Daloze)

Right, @nobu's approach seems much better than reintroducing that weird behavior for .dup.

Ideally we wouldn't rehash as in calling key.hash methods again, but instead just shrink the internal data structure (and same when growing it).

#4 - 10/24/2023 03:05 PM - Eregon (Benoit Daloze)

So apparently some applications were relying on Set#dup/Hash#dup to do like C++ [shrink_to_fit](#).

Ruby does not have such a method and it feels quite low-level, so it seems better to resize the internal data structure when removing elements/entries and going below some threshold.

#5 - 11/08/2023 10:29 AM - Eregon (Benoit Daloze)

As a note, this repro code is very "lucky" to trigger a dup after removing 99.99% of the elements.

I suppose it's done that way to make the effect very clear though.

Without the - [0] the same problem occurs on 3.0:

```
$ ruby -v -rset -e 's1 = Set.new(10000.times); s2 = Set.new(9999.times); a=Array.new(10000) { s1 - s2 }; GC.start; puts `ps -o rss= -p #{$$}`.to_i'
ruby 3.0.6p216 (2023-03-30 revision 23a532679b) [x86_64-linux]
3015808
$ ruby -v -rset -e 's1 = Set.new(10000.times); s2 = Set.new(9999.times); a=Array.new(10000) { s1 - s2 - [0] }; GC.start; puts `ps -o rss= -p #{$$}`.to_i'
ruby 3.0.6p216 (2023-03-30 revision 23a532679b) [x86_64-linux]
74552
```

If a Set is kept alive a long time, one way to ensure it uses the minimum amount of space is `Set#reset`, at the cost of extra time to reset/rehash (which notably calls `#hash` for every key), it's a time vs memory trade-off, can be worth it for big long-lived sets:

```
$ ruby -v -rset -e 's1 = Set.new(10000.times); s2 = Set.new(9999.times); a=Array.new(10000) { s=s1 - s2 - [0]; s.reset; s }; GC.start; puts `ps -o rss= -p #{$$}`.to_i'
ruby 3.2.2 (2023-03-30 revision e51014f9c0) [x86_64-linux]
62992
```

Automatic shrinking (PR at <https://github.com/ruby/ruby/pull/8748>) should help the worst cases like the repro so that seems good anyway.

#6 - 11/11/2023 10:24 AM - nobu (Nobuyoshi Nakada)

- Status changed from Open to Closed

Applied in changeset [git|9eac9d71786a8dbec520d0541a91149f01adf8ea](https://github.com/ruby/ruby/commit/9eac9d71786a8dbec520d0541a91149f01adf8ea).

[Bug [#19969](#)] Compact `st_table` after deleted if possible

#7 - 11/19/2023 01:03 PM - nagachika (Tomoyuki Chikanaga)

- Backport changed from 3.0: DONTNEED, 3.1: REQUIRED, 3.2: REQUIRED to 3.0: DONTNEED, 3.1: REQUIRED, 3.2: DONE

ruby_3_2 1cc38d5a2f84733e1c2e42548639e2891fe61e69 merged revision(s) 9eac9d71786a8dbec520d0541a91149f01adf8ea.

#8 - 11/20/2023 02:51 AM - hsb (Hiroshi SHIBATA)

Thanks nobu and nagachika.

I confirmed to resolve this regresssion with `ruby_3_2` branch.

```
# Before
$ ruby -v -rset -e 's1 = Set.new(10000.times); s2 = Set.new(9999.times); Array.new(10000) { s1 - s2 - [0] }; puts `ps -o rss= -p #{$$}`.to_i'
ruby 3.2.2 (2023-07-05 revision 2f603bc4d7) +YJIT [arm64-darwin23]
4564304
```

```
# After
$ ruby -v -rset -e 's1 = Set.new(10000.times); s2 = Set.new(9999.times); Array.new(10000) { s1 - s2 - [0] }; puts `ps -o rss= -p #{$$}`.to_i'
ruby 3.2.2 (2023-11-19 revision d9f4f321c6) +YJIT [arm64-darwin23]
40864
```

#9 - 11/20/2023 12:11 PM - usa (Usaku NAKAMURA)

- Backport changed from 3.0: DONTNEED, 3.1: REQUIRED, 3.2: DONE to 3.0: DONTNEED, 3.1: DONE, 3.2: DONE

ruby_3_1 1cae5e7ceaca7304108fdec35d4858a9e4ff7fe0 merged revision(s) 9eac9d71786a8dbec520d0541a91149f01adf8ea.