Ruby trunk - Feature #9633

[PATCH]Symbol GC

03/13/2014 07:59 AM - authorNari (Narihiro Nakamura)

Status:	Closed
Priority:	Normal
Assignee:	matz (Yukihiro Matsumoto)
Target version:	2.2.0

Description

I've written a patch to collect most symbols.

PATCH: https://github.com/authorNari/ruby/compare/13834fb3c...symbol_gc.patch

Summary

- Most symbols in Ruby level are GC-able generated by #to_sym, #intern, etc..
- Exclude a symbol which is translated ID in C-level from GC-able symbols
- Keep Ruby's C extension compatibility
- Pass make test-all

Benchmark

A benchmark program is here.

```
obj = Object.new
100_000.times do |i|
  obj.respond_to?("sym#{i}".to_sym)
end
GC.start
puts"symbol : #{Symbol.all_symbols.size}"
```

```
% time RBENV_VERSION=ruby-r45059 ruby -v /tmp/a.rb
ruby 2.2.0dev (2014-02-20 trunk 45059) [x86_64-linux]
symbol : 102416
0.24s user 0.01s system 91% cpu 0.272 total

% time RBENV_VERSION=symgc ruby -v /tmp/a.rb
ruby 2.2.0dev (2014-02-20 trunk 45059) [x86_64-linux]
symbol : 2833
0.21s user 0.01s system 90% cpu 0.247 total
```

The total number of symbols is declined.
The total time of symgc version is improved because Full GC pressure has been reduced.

The result of make benchmark.

https://gist.github.com/authorNari/9359704

There is no significant slowdown.

(I would welcome to try an additional benchmark and report)

Implementation Detail

I classify Dynamic symbol and Static symbol.

- Static symbol

    - Generated by rb_itnern()
    - A sequential unique number as in the past.
    - Not GC-able
    - LSB = 1
    - Reserved IDs(147 and below) are exceptional cases

- Dynamic symbol

    - Generated by #to_sym, #intern in Ruby level
    - RVALUE
    - GC-able
    - LSB = 0
    - Pin down a dynamic symbol when it translate to ID (e.g. SYM2ID, rb_intern).
    - Pinned dynamic symbols are never collected.
    - I'd like to include ID in GC's roots only CRuby internal in order to reduce pinned dynamic symbols.

Please read the patch if you want to know more information.

Acknowledgment

The idea of this symbol GC is invented by Sasada Koichi in Heroku,inc.
Thank you.

-- ja --
Ruby□□□□□□□□□GC□□□□□□□□□□□□□□□□
https://github.com/authorNari/ruby/compare/13834fb3c...symbol_gc

□ □

- Ruby□□□□□□□□□□□□□□□GC□□□to_sym,intern□□□□□□□□
- C□□ID□□□□□□□□□□GC□□□□□□□□rb_intern□SYM2ID□□□
- C-API□□□□□□
- make test-all□□□

□ □ □ □ □ □

□□□□□□□□□□□□□

```
obj = Object.new
100_000.times do |i|
  obj.respond_to?("sym#{i}".to_sym)
end
GC.start
puts"symbol : #{Symbol.all_symbols.size}"
```

```
% time RBENV_VERSION=symgc ruby -v /tmp/a.rb
ruby 2.2.0dev (2014-02-20 trunk 45059) [x86_64-linux]
symbol : 2833
0.21s user 0.01s system 90% cpu 0.247 total

% time RBENV_VERSION=ruby-r45059 ruby -v /tmp/a.rb
ruby 2.2.0dev (2014-02-20 trunk 45059) [x86_64-linux]
symbol : 102416
0.24s user 0.01s system 91% cpu 0.272 total
```

□□□□□□□□□□□□□□□□□□□□□□
□□□□□□□□□□Full GC□□□□□□□□□□□□□□□□□□□□□symgc□□□□□□□□□□

make benchmark□□□□
https://gist.github.com/authorNari/9359704

□□□□□□□□□□□□□□□

□□□□□□□□□□□□□□□□

□ □ □ □ □ □ □ □ □ □

symbol□static symbol□dynamic symbol□□□□

- static symbol
  - rb_itnern□□□□□□□□□□□
  - □□□□□□□□□□□□□□
  - GC□□□
  - □□1□□□□□□□□□□1□□□□
  - 147□□□□□□□□ID□□□□□□□

- dynamic symbol
  - Ruby□□□□#to_sym,#intern□□□□□□□□□□□
  - RVALUE□□□□□
  - GC□□
  - □□1□□□□0
  - C□□□□ID□□□SYM2ID□□□□□□□□□□pindown□□GC□□□□□□□□□□
  - Ruby□□□ID□□□□□□□□□pindown□□□□□□□□□□□

□□□□□□□□□□□□□□□□□□□□□…□

□ □

□□□□GC□□□□□□□Heroku□□□□□□□□□□□□□□□□□□□□
□□□□□□□□□□□□

History

#1 - 03/13/2014 08:05 AM - authorNari (Narihiro Nakamura)

- Status changed from Open to Closed

□□□□□□ ruby-dev□□□□□□□□□□□
#9634 □□□□□…□