Ruby master - Feature #16505
Improve performance of `RubyVM::InstructionSequence#to_binary`

01/12/2020 01:42 PM - NagayamaRyoga (Nagayama Ryoga)

Status: Closed
Priority: Normal
Assignee:
Target version:

Description

Abstract

Within #to_binary, deduplication of objects output to binary is performed, but the current implementation is achieved by a linear search of an array of objects (=obj_list). (https://github.com/ruby/ruby/blob/e288632f22b18b29efd20a1469232b0a3ba9b74c/compile.c#L9699-L9701)

On the other hand, iseq deduplication is faster because it is implemented using a hash. (https://github.com/ruby/ruby/blob/e288632f22b18b29efd20a1469232b0a3ba9b74c/compile.c#L9744-L9745)

This proposal speeds up object deduplication by using a hash.
This patch does not change the output binary.

Implementation

https://github.com/ruby/ruby/pull/2835

Evaluation

Environment:

- OS: macOS Catalina
- CPU: Intel Core i5
- Memory: 16GB

address_lists_parser.rb

address_lists_parser.rb (https://github.com/mikel/mail/blob/master/lib/mail/parsers/address_lists_parser.rb) in mail gem has an extremely huge array.
Call #to_binary on the iseq of this file and check its execution time and MD5 of the output binary.

The benchmark code:

```ruby
require 'benchmark'
require 'digest/md5'

F = 'address_lists_parser.rb'
N = 100

iseq = RubyVM::InstructionSequence.compile_file(F)
bin = iseq.to_binary

puts "md5 hash: #{Digest::MD5.hexdigest(bin)}"

Benchmark.bm(12) do |x|
  x.report("to_binary x#{(N)}") do
    N.times do |i|
      iseq.to_binary
    end
  end
end
```

- master (ruby 2.8.0dev (2020-01-12T10:54:59Z master e288632f22) [x86_64-darwin19])
The same binary was output before and after the change. Execution speed is 26 times faster.

---

**Associated revisions**

**Revision 6e5e6a40** - 02/09/2020 04:33 PM - NagayamaRyoga (Nagayama Ryoga)
Deduplicate objects efficiently when dumping iseq to binary

We were inefficient in cases where there are a lot of duplicates due to the use of linear search. Use a hash table instead.

These cases are not that rare in the wild.

[Feature #16505]

**Revision e443f235** - 02/09/2020 04:33 PM - NagayamaRyoga (Nagayama Ryoga)
compile.c: Drop iseq_list from ibf_dump

[Feature #16505]

**Revision a2845adb** - 02/09/2020 04:33 PM - NagayamaRyoga (Nagayama Ryoga)
compile.c: Drop obj_list from ibf_dump

[Feature #16505]

**History**

#1 - 02/09/2020 04:33 PM - NagayamaRyoga (Nagayama Ryoga)
- Status changed from Open to Closed

Applied in changeset git|6e5e6a40c4c35aee1cfb7d0e8a947354f80baa9e.

Deduplicate objects efficiently when dumping iseq to binary

We were inefficient in cases where there are a lot of duplicates due to the use of linear search. Use a hash table instead.

These cases are not that rare in the wild.

[Feature #16505]