

## Ruby master - Bug #9382

### [patch] add opt\_aref\_str and opt\_aset\_str

01/08/2014 06:07 PM - tmm1 (Aman Gupta)

<b>Status:</b>	Closed	
<b>Priority:</b>	Normal	
<b>Assignee:</b>	tmm1 (Aman Gupta)	
<b>Target version:</b>	2.2.0	
<b>ruby -v:</b>	trunk	<b>Backport:</b> 1.9.3: DONTNEED, 2.0.0: DONTNEED, 2.1: DONTNEED

#### Description

[charliesome \(Charlie Somerville\)](#), [normalperson \(Eric Wong\)](#) and I have been working on a patch to add two new instructions:

<https://bugs.ruby-lang.org/issues/9188#note-9>

These optimize Hash#[] and Hash#[]=, when used with a string literal key.

Before:

```
0010 putstring    "str"
0012 opt_aref
```

After:

```
0010 opt_aref_str    , "str"
```

The opt\_(aref|aset)\_str instructions avoid str\_duplicate, resulting in reduced allocation overhead. For example, this loop requires zero allocations after the patch:

```
hash={}; loop{ hash["key"] += 1 }
```

In our rails app, 2.5% of iseqs can benefit from these optimizations:

```
$ ruby -rconfig/environment -e' p ObjectSpace.each_object(RubyVM::InstructionSequence).count '
63314
```

```
$ ruby -rconfig/environment -e' p ObjectSpace.each_object(RubyVM::InstructionSequence).select{ |i| i.disasm.match(/
opt_(aref|aset)_str /) }.size '
1464
```

[headius \(Charles Nutter\)](#) is also adding a similar optimization to jruby: <https://bugs.ruby-lang.org/issues/8998#note-15>

I plan to commit the patch above, so if you have any feedback on the implementation please share.

#### Related issues:

Related to Ruby master - Feature #8998: string keys for hash literals should ...	<b>Closed</b>	<b>10/08/2013</b>
Related to Ruby master - Misc #9188: r43870 make benchmark/bm_so_k_nucleotide...	<b>Closed</b>	<b>12/01/2013</b>
Related to Ruby master - Bug #12855: Inconsistent keys identity in compare_by...	<b>Closed</b>	

#### Associated revisions

##### Revision 58f800a2 - 01/10/2014 04:54 AM - tmm1 (Aman Gupta)

insns.def: add opt path for Hash#[] and Hash#[]= used with str literal keys

- insns.def (opt\_aref\_with): new instruction to optimize Hash#[], removing any allocation overhead when used with a string literal key. Patch by normalperson (Eric Wong). [ruby-core:59640] [Bug #9382]
- insns.def (opt\_aset\_with): new instruction to optimize Hash#[]=
- compile.c (iseq\_compile\_each): compiler shortcuts for new instructions
- hash.c (static VALUE rb\_hash\_compare\_by\_id\_p): fix documentation for Hash#compare\_by\_identity to reflect frozen string sharing
- test/ruby/test\_hash.rb (class TestHash): test for new behavior

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@44551 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

#### Revision 44551 - 01/10/2014 04:54 AM - tmm1 (Aman Gupta)

insns.def: add opt path for Hash#[] and Hash#[]= used with str literal keys

- insns.def (opt\_aref\_with): new instruction to optimize Hash#[], removing any allocation overhead when used with a string literal key. Patch by normalperson (Eric Wong). [ruby-core:59640] [Bug #9382]
- insns.def (opt\_aset\_with): new instruction to optimize Hash#[]=
- compile.c (iseq\_compile\_each): compiler shortcuts for new instructions
- hash.c (static VALUE rb\_hash\_compare\_by\_id\_p): fix documentation for Hash#compare\_by\_identity to reflect frozen string sharing
- test/ruby/test\_hash.rb (class TestHash): test for new behavior

#### Revision 44551 - 01/10/2014 04:54 AM - tmm1 (Aman Gupta)

insns.def: add opt path for Hash#[] and Hash#[]= used with str literal keys

- insns.def (opt\_aref\_with): new instruction to optimize Hash#[], removing any allocation overhead when used with a string literal key. Patch by normalperson (Eric Wong). [ruby-core:59640] [Bug #9382]
- insns.def (opt\_aset\_with): new instruction to optimize Hash#[]=
- compile.c (iseq\_compile\_each): compiler shortcuts for new instructions
- hash.c (static VALUE rb\_hash\_compare\_by\_id\_p): fix documentation for Hash#compare\_by\_identity to reflect frozen string sharing
- test/ruby/test\_hash.rb (class TestHash): test for new behavior

#### Revision 44551 - 01/10/2014 04:54 AM - tmm1 (Aman Gupta)

insns.def: add opt path for Hash#[] and Hash#[]= used with str literal keys

- insns.def (opt\_aref\_with): new instruction to optimize Hash#[], removing any allocation overhead when used with a string literal key. Patch by normalperson (Eric Wong). [ruby-core:59640] [Bug #9382]
- insns.def (opt\_aset\_with): new instruction to optimize Hash#[]=
- compile.c (iseq\_compile\_each): compiler shortcuts for new instructions
- hash.c (static VALUE rb\_hash\_compare\_by\_id\_p): fix documentation for Hash#compare\_by\_identity to reflect frozen string sharing
- test/ruby/test\_hash.rb (class TestHash): test for new behavior

#### Revision 44551 - 01/10/2014 04:54 AM - tmm1 (Aman Gupta)

insns.def: add opt path for Hash#[] and Hash#[]= used with str literal keys

- insns.def (opt\_aref\_with): new instruction to optimize Hash#[], removing any allocation overhead when used with a string literal key. Patch by normalperson (Eric Wong). [ruby-core:59640] [Bug #9382]
- insns.def (opt\_aset\_with): new instruction to optimize Hash#[]=
- compile.c (iseq\_compile\_each): compiler shortcuts for new instructions
- hash.c (static VALUE rb\_hash\_compare\_by\_id\_p): fix documentation for Hash#compare\_by\_identity to reflect frozen string sharing
- test/ruby/test\_hash.rb (class TestHash): test for new behavior

#### Revision 44551 - 01/10/2014 04:54 AM - tmm1 (Aman Gupta)

insns.def: add opt path for Hash#[] and Hash#[]= used with str literal keys

- insns.def (opt\_aref\_with): new instruction to optimize Hash#[], removing any allocation overhead when used with a string literal key. Patch by normalperson (Eric Wong). [ruby-core:59640] [Bug #9382]
- insns.def (opt\_aset\_with): new instruction to optimize Hash#[]=
- compile.c (iseq\_compile\_each): compiler shortcuts for new instructions
- hash.c (static VALUE rb\_hash\_compare\_by\_id\_p): fix documentation for Hash#compare\_by\_identity to reflect frozen string sharing
- test/ruby/test\_hash.rb (class TestHash): test for new behavior

#### Revision 44551 - 01/10/2014 04:54 AM - tmm1 (Aman Gupta)

insns.def: add opt path for Hash#[] and Hash#[]= used with str literal keys

- insns.def (opt\_aref\_with): new instruction to optimize Hash#[], removing any allocation overhead when used with a string literal key. Patch by normalperson (Eric Wong). [ruby-core:59640] [Bug #9382]
- insns.def (opt\_aset\_with): new instruction to optimize Hash#[]=
- compile.c (iseq\_compile\_each): compiler shortcuts for new instructions
- hash.c (static VALUE rb\_hash\_compare\_by\_id\_p): fix documentation for Hash#compare\_by\_identity to reflect frozen string sharing
- test/ruby/test\_hash.rb (class TestHash): test for new behavior

## History

---

#### #1 - 01/08/2014 08:07 PM - tmm1 (Aman Gupta)

With the patch, "allocations per request" in our app is reduced by ~3%.

#### #2 - 01/29/2014 09:20 PM - nobu (Nobuyoshi Nakada)

- Backport changed from 1.9.3: UNKNOWN, 2.0.0: UNKNOWN, 2.1: UNKNOWN to 1.9.3: DONTNEED, 2.0.0: DONTNEED, 2.1: DONTNEED

- Status changed from Open to Closed

**#3 - 12/21/2016 02:47 PM - shyouhei (Shyouhei Urabe)**

- Related to Bug #12855: Inconsistent keys identity in compare\_by\_identity Hash when using literals added