

## Ruby master - Bug #16950

### Stop nonsense keyword argument warnings in 2.6

06/11/2020 07:14 AM - mame (Yusuke Endoh)

<b>Status:</b>	Open		
<b>Priority:</b>	Normal		
<b>Assignee:</b>	usa (Usaku NAKAMURA)		
<b>Target version:</b>			
<b>ruby -v:</b>	ruby 2.6.7p147 (2020-03-31 revision 67883) [x86_64-linux]	<b>Backport:</b>	2.5: UNKNOWN, 2.6: UNKNOWN, 2.7: UNKNOWN

#### Description

Ruby 2.6 warns the following code with `-w` option:

```
def foo(x)      # warning: in `foo': the last argument was passed as a single Hash
end
h = { k: 42 }
foo(**h)       # warning: although a splat keyword arguments here
```

This warning had been introduced by [nobu \(Nobuyoshi Nakada\)](#) based on the original proposal of [#14183 "Real" keyword argument](#). However, the design was changed so that automatic conversion from keywords to positional arguments will be kept in 3.0. In other words, the code above will work even in 3.0, so the warning no longer makes sense.

Rather, according to [matsuda \(Akira Matsuda\)](#) and [kamipo \(Ryuta Kamizono\)](#), this warning sometimes hindered the fix work for the keyword argument change in Rails. This issue is possible to work around by using always non-keyword argument (and [they actually did so](#)), but they say it is very annoying.

So, I propose stopping the warning in 2.6. This is a change for 2.6, but matz already accepted this. [usa \(Usaku NAKAMURA\)](#), do you accept this?

A patch is quite simple:

```
diff --git a/vm_args.c b/vm_args.c
index 299ed16aa8..2e70ed3fcf 100644
--- a/vm_args.c
+++ b/vm_args.c
@@ -685,17 +685,6 @@
  setup_parameters_complex(rb_execution_context_t * const ec, const rb_iseq_t * co
    else if (!NIL_P(keyword_hash) && RHASH_SIZE(keyword_hash) > 0) {
      argument_kw_error(ec, iseq, "unknown", rb_hash_keys(keyword_hash));
    }
-   else if (kw_splat && NIL_P(keyword_hash)) {
-     if (RTEST(ruby_verbose)) {
-       VALUE path = rb_iseq_path(iseq);
-       VALUE line = rb_iseq_first_lineno(iseq);
-       VALUE label = rb_iseq_label(iseq);
-       rb_compile_warning(NIL_P(path) ? NULL : RSTRING_PTR(path), FIX2INT(line),
-         "in `%s': the last argument was passed as a single Hash",
-         NIL_P(label) ? NULL : RSTRING_PTR(label));
-       rb_warning("although a splat keyword arguments here");
-     }
-   }

  if (iseq->body->param.flags.has_block) {
    if (iseq->body->local_iseq == iseq) {
```

#### Related issues:

Is duplicate of Ruby master - Bug #16632: Remove verbose warning on treating ...

Closed

#### History

**#1 - 06/11/2020 09:11 AM - kamipo (Ryuta Kamizono)**

Using non-keyword argument is not always workaround for that.

For example, redis 4.2.0 gem has changed options hash to kwargs.

To address kwargs warnings, we need to add \*\* to the method call, it works for both redis 4.2.0 and 4.1.4 with using Ruby 2.7.

But if we use Ruby 2.6, adding \*\* works for redis 4.2.0, but raise the warning for redis 4.1.4.

<https://github.com/redis/redis-rb/blob/v4.2.0/lib/redis.rb#L816>

<https://github.com/redis/redis-rb/blob/v4.1.4/lib/redis.rb#L790>

Strictly speaking, there is a way to avoid the warning by checking RUBY\_VERSION (and gem version).

But I've finally given up to address the warning in the Rails codebase.

<https://github.com/rails/rails/commit/37c19f7ebcf542af968a4983b3296b9fa283a0dc#diff-60aba42516f795d80ed6f18ad14c1f6aR401>

**#2 - 06/11/2020 02:05 PM - jeremyevans0 (Jeremy Evans)**

- *Is duplicate of Bug #16632: Remove verbose warning on treating keyword splat as positional argument in Ruby 2.6 added*

**#3 - 06/18/2020 08:24 AM - usa (Usaku NAKAMURA)**

I see, I'll patch later.