

Ruby master - Feature #12913

A way to configure the default maximum width of pp

11/09/2016 03:04 PM - mame (Yusuke Endoh)

| | |
|---|--------------------|
| Status: | Rejected |
| Priority: | Normal |
| Assignee: | akr (Akira Tanaka) |
| Target version: | |
| Description | |
| How about having an easy way to configure the maximum width of a line of pp output? Currently, pp accepts the maximum width as an optional argument: | |
| <pre>pp(big_array, \$>, 120)</pre> | |
| However, this is obviously too long for a useful debugging-purpose method like pp. Even worse, we must add the fragment ", \$>, 120" to all calls to pp. I don't feel this is reasonable. | |
| The patch attached provides PP.default_maxwidth= and PP.default_maxwidth, which can be used to configure the default setting of the maxwidth. | |
| <pre>PP.default_maxwidth = 1 pp([1, 2, 3]) #=> [1, # 2, # 3]</pre> | |
| Akr-san, what do you think? | |

History

#1 - 11/09/2016 04:10 PM - akr (Akira Tanaka)

I think the columns obtained by IO#winsize of io/console is better default now.

I'm not sure you are satisfied with it, though.

#2 - 11/09/2016 10:52 PM - mame (Yusuke Endoh)

Akr-san, thank you for your reply.

I agree that it is a better default than the fixed 79.

However, I'd still like to have PP.default_maxwidth= too.
My motivation is to show abstract syntax tree (in [my article](#)).
In such a case, I think that "break as far as possible" would be useful.

Thank you for your consideration,

#3 - 11/11/2016 02:04 AM - akr (Akira Tanaka)

- Status changed from Open to Feedback

Yusuke Endoh wrote:

In such a case, I think that "break as far as possible" would be useful.

"break as far as possible" is opposite to pp.rb's intent:
"don't break as far as possible (in screen width limitation)".

I think a global configuration to choose different intent for single API is not a good idea.

pp (including pretty_inspect and PP.pp) method is used in many gems.
I think the intent of most of them is "don't break as far as possible".
PP.default_maxwidth = 1 breaks the intent.

I recommend that you define a method to call PP.pp with width = 1.

#4 - 11/11/2016 03:31 AM - mame (Yusuke Endoh)

Akira Tanaka wrote:

pp (including pretty_inspect and PP.pp) method is used in many gems.

Note that my proposal is only for PP; the scope does not include pretty_inspect.

PP is a debugging-purpose API, so it is rarely used in a released code. So, we don't have to consider the case where PP is used in multiple modules simultaneously, I think. Also, because of debugging-purpose, it would be good to prioritize usability than safety.

#5 - 12/21/2016 07:15 AM - akr (Akira Tanaka)

I think pp is not only for debug.

I guess someone use it for logging, for example.

So, I think global configuration is dangerous,

#6 - 12/21/2016 09:06 AM - mame (Yusuke Endoh)

- Status changed from Feedback to Rejected

Okay, I respect your decision. Thank you for your time.

#7 - 02/06/2018 04:47 AM - nobu (Nobuyoshi Nakada)

What about this?

- affects PP.pp and Kernel#pp only
- try console window size, COLUMNS environment variable, then old good 80

```
diff --git a/lib/pp.rb b/lib/pp.rb
indiff --git a/lib/pp.rb b/lib/pp.rb
index 85401c8aa6..a364d0fc2a 100644
--- a/lib/pp.rb
+++ b/lib/pp.rb
@@ -68,7 +68,7 @@
  # If +width+ is omitted, 79 is assumed.
  #
  # PP.pp returns +out+.
- def PP.pp(obj, out=$>, width=79)
+ def PP.pp(obj, out=$>, width=PP.width_for(out))
  q = PP.new(out, width)
  q.guard_inspect_key {q.pp obj}
  q.flush
@@ -91,6 +91,15 @@
  def PP.mcall(obj, mod, meth, *args, &block)
    mod.instance_method(meth).bind(obj).call(*args, &block)
  end
+
+ def PP.width_for(out)
+   begin
+     require 'io/console'
+     _, width = out.winsize
+     rescue LoadError, NoMethodError, Errno::ENOTTY
+     end
+   (width || ENV['COLUMNS']&.to_i&.nonzero? || 80) - 1
+ end
  # :startdoc:

  @sharing_detection = false
@@ -549,9 +558,9 @@
  # prints arguments in pretty form.
  #
  # pp returns argument(s).
- def pp(*objs)
+ def pp(*objs, out: $>, width: PP.width_for(out))
  objs.each {|obj|
-   PP.pp(obj)
+   PP.pp(obj, out, width)
  }
  objs.size <= 1 ? objs.first : objs
end
```

#8 - 02/06/2018 05:30 AM - akr (Akira Tanaka)

nobu (Nobuyoshi Nakada) wrote:

What about this?

- affects PP.pp and Kernel#pp only
- try console window size, COLUMNS environment variable, then old good 80

It seems fine.

Files

| | | | |
|---------------------------|---------|------------|---------------------|
| pp-default-maxwidth.patch | 1.05 KB | 11/09/2016 | mame (Yusuke Endoh) |
|---------------------------|---------|------------|---------------------|