

Ruby - Feature #20211

Consider re-adding 3.2-style support of Anonymous Args/Blocks

01/25/2024 05:00 AM - johnnyshields (Johnny Shields)

<div>Status:Open</div> <div>Priority:Normal</div> <div>Assignee:</div> <div>Target version:</div>	
<div>Description</div> <div>(My sincere apologies if this issue has already been raised, I didn't see it in Ruby 3.4 backlog.)</div> <div>As per <a href="https://bugs.ruby-lang.org/issues/19370">https://bugs.ruby-lang.org/issues/19370</a>, Ruby 3.3 no longer allows the following:</div> <div><pre>def m(*)   [1, 2, 3].each {  *  p(*) } end m('test', 'test', 'test') #=&gt; Ruby 3.2: p(*) uses args from m(*) #=&gt; test, test, test #=&gt; Ruby 3.3: raises SyntaxError: anonymous rest parameter is also used within block</pre></div> <div>I am concerned because Rubocop auto-corrects Ruby 3.1+ code to syntax which is no longer supported in Ruby 3.3. There may be quite a bit of Ruby code in the wild which breaks on 3.3 (my own app broke in about 50 places).</div> <div><ul style="list-style-type: none"><li><a href="#">Ruby 3.1+: Naming/BlockForwarding (default true)</a></li><li><a href="#">Ruby 3.2+: Style/ArgumentsForwarding with UseAnonymousForwarding (default true)</a></li></ul></div> <div>It would be nice if the Ruby 3.4 syntax could allow the old nested syntax, supporting arg shadowing with different behavior between implicit and explicit <code> * </code> usage.</div> <div><pre>def m(*)   [1, 2, 3].each { p(*) } end m('test', 'test', 'test') #=&gt; Ruby 3.2: p(*) uses args from m(*) #=&gt; test, test, test #=&gt; Ruby 3.4: p(*) uses args from m(*) #=&gt; test, test, test  def m(*)   [1, 2, 3].each {  *  p(*) } end m('test', 'test', 'test') #=&gt; Ruby 3.2: p(*) uses args from m(*) #=&gt; test, test, test #=&gt; Ruby 3.4: p(*) uses args from each #=&gt; 1, 2, 3</pre></div> <div>The same would also apply to anonymous blocks.</div>	

History

#1 - 01/25/2024 05:00 AM - johnnyshields (Johnny Shields)

- Subject changed from Reconsider introducing 3.2-style support of Anonymous Args/Blocks to Consider re-adding 3.2-style support of Anonymous Args/Blocks