

Ruby - Misc #15012

Testing on latest version gcc on Travis CI

08/20/2018 02:23 PM - jaruga (Jun Aruga)

Status:	Closed	
Priority:	Normal	
Assignee:		
Description		
Ruby's Travis CI [1] is running on gcc 4.8.4 as the default behavior of Travis gcc setting. Right now gcc's latest version is 8. Though it is important to test on the old version gcc-4.8, also it is meaningful to test on the latest version gcc, knowing we have RubyCI. [2] The benefit to run on gcc-8 is • We can care the issue on the latest version of gcc easier, such as visualizing the build warnings on the log. • We can show contributors or developers how ruby could be built on the latest version gcc. I sent the pull-request [3] to show the code as a proposal. [1] https://travis-ci.org/ruby/ruby [2] https://rubyci.org/ [3] https://github.com/ruby/ruby/pull/1937		

History

#1 - 08/20/2018 02:30 PM - jaruga (Jun Aruga)

The demerit is

- The Travis test running time could be longer than current one. But maybe not so long time because those test cases are run in parallel.
- I am not sure whether the "ppa:ubuntu-toolchain-r/test" repository [1] shown on the Travis document [2] is stable.

[1] <https://launchpad.net/~ubuntu-toolchain-r/+archive/ubuntu/test>

[2] <https://docs.travis-ci.com/user/installing-dependencies/#installing-packages-from-a-custom-apt-repository>

#2 - 08/20/2018 03:21 PM - jaruga (Jun Aruga)

If you want to imagine how gcc-N and clang test cases work on both linux and osx of Travis CI, I can share this project's case.

<https://travis-ci.org/trinityrnaseq/trinityrnaseq>

<https://github.com/trinityrnaseq/trinityrnaseq/blob/master/.travis.yml>

#3 - 08/21/2018 12:41 AM - MSP-Greg (Greg L)

Given that Appveyor/Windows MSYS2/MinGW also has gcc 8.2.0 available (ruby-loco uses it), seems like a good idea to me. One would hope that newer gcc releases are also better optimized for JIT...

Off topic - about Appveyor testing - MSYS2/MinGW - mingw-w64-x86_64-toolchain is gcc. There is also a base-devel group. ruby-loco updates both, but toolchain is probably all that is required...

#4 - 08/21/2018 10:04 AM - jaruga (Jun Aruga)

Given that Appveyor/Windows MSYS2/MinGW also has gcc 8.2.0 available (ruby-loco uses it), seems like a good idea to me. One would hope that newer gcc releases are also better optimized for JIT...

It looks good idea to add gcc 8 on the Windows environment too.

But I need someone's help for that, as I am not familiar with it.

I can work for Travis osx case if it is needed.

#5 - 08/21/2018 01:44 PM - MSP-Greg (Greg L)

[@jaruga \(Jun Aruga\)](#).

Sorry, my message was unclear. For quite some time, Ruby trunk has been built on Appveyor using the MSFT VC compiler (mswin). Sometime last

year, full testing was added to that build.

In the wild, most Windows Ruby users run builds based on MinGW based compilers. Older builds used MSYS/MinGW, newer builds use MSYS2/MinGW. These are often referred to as RubyInstaller builds (the newer ones are technically RubyInstaller2).

Separate from CI here, Ruby trunk has also been built and fully tested using the MSYS2/MinGW tools on Appveyor.

Here, MSYS2/MinGW building/testing was recently added to the Appveyor CI, so the CI is now testing both mswin and mingw builds.

I was trying to point out that the MSYS/MinGW toolset is also currently using gcc 8.2.0, but the Appveyor tools, being somewhat out of date, are currently at 7.3.0. They can be updated in the CI...

Thanks, Greg

#6 - 08/22/2018 11:47 AM - jaruga (Jun Aruga)

Hi Greg,

Sorry, my message was unclear.

No problem at all. Thanks for the explanation to be cleared.

the CI is now testing both mswin and mingw builds.

So, adding gcc-8 to the both environments makes sense for me.

What I want to share as one of the benefits is

- It is beneficial for people who are maintaining the Debian or RPM package of Ruby or gem packages on Debian or Fedora project like me. :) The latest version gcc is actively used as a main compiler on Fedora project.

#7 - 09/16/2018 12:04 PM - Anonymous

I just noticed the environment variables line for gcc-8 test case on .travis.yml are removed by the great hack in configure.ac.

That's amazing!

Thank you for the improvement.

<https://github.com/ruby/ruby/commit/f677ba8>

```
$ git show f677ba887158758c48b4f8e1df092c6aacfe6621
...
-     env:
-       - CC=gcc-8 AR=gcc-ar-8 NM=gcc-nm-8 RANLIB=gcc-ranlib-8
...
```

#8 - 05/25/2021 01:13 PM - jaruga (Jun Aruga)

- Status changed from Open to Closed

I would close this ticket, as the gcc latest version is tested on the GitHub Actions compilers.yml now.

<https://github.com/ruby/ruby/blob/master/.github/workflows/compilers.yml>