

Ruby - Bug #13654

irb save-history extension is not concurrency-safe

06/13/2017 01:17 PM - charles.cooke (Charles Cooke)

Status:	Closed	
Priority:	Normal	
Assignee:	aycabta (aycabta .)	
Target version:		
ruby -v:	ruby 2.4.1p111 (2017-03-22 revision 58053) [x86_64-linux]	Backport: 2.2: UNKNOWN, 2.3: UNKNOWN, 2.4: UNKNOWN

Description

Issue

If 2 simultaneous irb sessions are running with save-history enabled, the history will be overwritten in full by the last irb session to exit and history will be lost.

The save-history extension currently loads the existing file on session start, maintains history throughout the session, then regenerates the history file with the preloaded + new history from the current session. Therefore if the file contents are changed while the irb session is running, those changes are not overwritten when the session exits.

As many systems have logrotate and could manage the file buffering independently, one idea would be to just append the history from current session, rather than recreating the history file.

Steps to reproduce

- 1. Enable the save-history extension to log to a file
- 2. Start an irb session (Session #1)
- 3. Start a second irb session (Session #2)
- 4. In Session #1, execute commands and exit - verify that the output is generated correctly
- 5. In Session #2, execute commands and exit - PROBLEM - the output from Session #1 has been deleted

See output below:

```
[current]$ cat .irbrc
require 'irb/ext/save-history'

IRB.conf[:SAVE_HISTORY] = 1000
IRB.conf[:HISTORY_FILE] = File.join(Dir.pwd, 'log', 'irb.log')

==== START SESSION #1# ====
[current]$ irb
irb(main):001:0> puts 'Session #1 - Output'
Session #1 - Output
=> nil
irb(main):002:0>
[1]+  Stopped                  irb
==== BACKGROUND SESSION #1# ====

==== START SESSION #2# ====
[current]$ irb
irb(main):001:0> puts 'Session #2 - Output'
Session #2 - Output
=> nil
irb(main):002:0> exit
==== STOP SESSION #2# ====

==== OUTPUT IS EXPECTED ====
[current]$ cat log/irb.log
puts 'Session #2 - Output'
exit
```

```
==== RESUME SESSION #1 ====
[current]$ fg

irb(main):003:0* puts 'Session #1 - Overwrites everything'
Session #1 - Overwrites everything
=> nil
irb(main):004:0> exit

==== PROBLEM - OUTPUT FROM SESSION #1 ONLY ====
[current]$ cat log/irb.log
puts 'Session #1 - Output'
puts 'Session #1 - Overwrites everything'
exit
```

Associated revisions

Revision 14e1739ff3ec81c9ea87a8aba03393f0bf0433a7 - 03/05/2021 03:18 PM - jeremyevans (Jeremy Evans)

[ruby/irb] Make save-history extension safe for concurrent use

This makes the save-history extension check for modifications to the history file before saving it. If the history file was modified after the history was loaded and before it was saved, append only the new history lines to the history file.

This can result in more lines in the history file than SAVE_HISTORY allows. However, that will be fixed the next time irb is run and the history is saved.

Fixes [Bug #13654]

<https://github.com/ruby/irb/commit/041ef53845>

Revision 14e1739ff3ec81c9ea87a8aba03393f0bf0433a7 - 03/05/2021 03:18 PM - jeremyevans (Jeremy Evans)

[ruby/irb] Make save-history extension safe for concurrent use

This makes the save-history extension check for modifications to the history file before saving it. If the history file was modified after the history was loaded and before it was saved, append only the new history lines to the history file.

This can result in more lines in the history file than SAVE_HISTORY allows. However, that will be fixed the next time irb is run and the history is saved.

Fixes [Bug #13654]

<https://github.com/ruby/irb/commit/041ef53845>

Revision 14e1739f - 03/05/2021 03:18 PM - jeremyevans (Jeremy Evans)

[ruby/irb] Make save-history extension safe for concurrent use

This makes the save-history extension check for modifications to the history file before saving it. If the history file was modified after the history was loaded and before it was saved, append only the new history lines to the history file.

This can result in more lines in the history file than SAVE_HISTORY allows. However, that will be fixed the next time irb is run and the history is saved.

Fixes [Bug #13654]

<https://github.com/ruby/irb/commit/041ef53845>

Revision cafa7904e7233d1a4b8ae7e738f6bd35ed67d642 - 04/03/2021 09:26 AM - aycabta (aycabta .)

Backport lib/reline, and lib/irb for 3.0.1 4th (#4349)

- [ruby/irb] Update help message for next context-mode of 4

While here, fixing tab/space issues in help message, and sync

rdoc for IRB class to match the help message.

<https://github.com/ruby/irb/commit/ef8e3901cc>

- [ruby/irb] Do not continue line if last expression is an endless range

Fixes [Bug #14824]

<https://github.com/ruby/irb/commit/63414f8465>

- [ruby/irb] Add a test for not continuing when endless range at eol

<https://github.com/ruby/irb/commit/1020ac9c65>

- [ruby/irb] Make save-history extension safe for concurrent use

This makes the save-history extension check for modifications to the history file before saving it. If the history file was modified after the history was loaded and before it was saved, append only the new history lines to the history file.

This can result in more lines in the history file than SAVE_HISTORY allows. However, that will be fixed the next time irb is run and the history is saved.

Fixes [Bug #13654]

<https://github.com/ruby/irb/commit/041ef53845>

- Fix errors when XDG_CONFIG_HOME points to non-writable directory

\$HOME/.config is not writable on CI
because I think tests should not corrupt user's data.

And GitHub Actions CI sets XDG_CONFIG_HOME
since Version: 20210309.1.

https://github.com/ruby/actions/runs/2130811016?check_suite_focus=true#step:16:301

```
Errno::EACCES: Permission denied @ dir_s_mkdir - /home/runner/.config/irb
```

- Try to fix errors in TestIRB::TestHistory too

https://github.com/ruby/actions/runs/2137935523?check_suite_focus=true#step:9:562

```
1) Error:
TestIRB::TestHistory#test_history_concurrent_use:
Errno::EACCES: Permission denied @ dir_s_mkdir - /home/runner/.config/irb
/home/runner/work/actions/actions/ruby/lib/fileutils.rb:253:in `mkdir'
/home/runner/work/actions/actions/ruby/lib/fileutils.rb:253:in `fu_mkdir'
/home/runner/work/actions/actions/ruby/lib/fileutils.rb:231:in `block (2 levels) in mkdir_p'
/home/runner/work/actions/actions/ruby/lib/fileutils.rb:229:in `reverse_each'
/home/runner/work/actions/actions/ruby/lib/fileutils.rb:229:in `block in mkdir_p'
/home/runner/work/actions/actions/ruby/lib/fileutils.rb:211:in `each'
/home/runner/work/actions/actions/ruby/lib/fileutils.rb:211:in `mkdir_p'
/home/runner/work/actions/actions/ruby/lib/irb/init.rb:355:in `rc_file_generators'
/home/runner/work/actions/actions/ruby/lib/irb/init.rb:330:in `rc_file'
/home/runner/work/actions/actions/ruby/test/irb/test_history.rb:170:in `block in assert_history'
/home/runner/work/actions/actions/ruby/lib/tmpdir.rb:96:in `mktmpdir'
/home/runner/work/actions/actions/ruby/test/irb/test_history.rb:168:in `assert_history'
/home/runner/work/actions/actions/ruby/test/irb/test_history.rb:133:in `test_history_concurrent_use'
```

- [ruby/irb] Define "measure" command without forced override

<https://github.com/ruby/irb/commit/9587ba13b5>

- [ruby/irb] Add all lib files automatically

<https://github.com/ruby/irb/commit/ecc82336b7>

- [ruby/irb] Don't call Ruby 2.4+'s String#pretty_print

<https://github.com/ruby/irb/commit/89bcf107be>

- [ruby/irb] Implement ls command

<https://github.com/ruby/irb/commit/19b6c20604>

- [ruby/irb] Add whereami command

<https://github.com/ruby/irb/commit/bc822e4aac>

- [ruby/irb] Fix column overflow on ls output

<https://github.com/ruby/irb/commit/6115754623>

- [ruby/irb] Fix step's argument

cols.size was calling Integer#size, which returns 8.

Fixing a bug of <https://github.com/ruby/irb/pull/209>

<https://github.com/ruby/irb/commit/c93ae4be71>

- [ruby/irb] Deal with different screen sizes

e.g. <http://rubyci.s3.amazonaws.com/centos8/ruby-master/log/20210321T063003Z.fail.html.gz>

<https://github.com/ruby/irb/commit/ddb3472ba2>

- [ruby/irb] Have some right padding

instead of filling out an entire line

<https://github.com/ruby/irb/commit/6ac8f45f5f>

- Suppress verbose messages

Get rid of warnings in the parallel test.

unknown command: "Switch to inspect mode."

- [ruby/irb] Change ripper_lex_without_warning to a class method

<https://github.com/ruby/irb/commit/d9f8abc17e>

- [ruby/irb] Complete require and require_relative

<https://github.com/ruby/irb/commit/1c61178b4c>

- [ruby/reline] Add Reline.ungetc to control buffer

<https://github.com/ruby/reline/commit/43ac03c624>

- [ruby/reline] Reline.delete_text removes the current line in multiline

<https://github.com/ruby/reline/commit/da90c094a1>

- [ruby/reline] Support preposing and postposing for Reline.completion_proc

<https://github.com/ruby/reline/commit/1f469de90c>

- [ruby/reline] Suppress crashing when completer_{quote,word_break}_characters is empty

<https://github.com/ruby/reline/commit/c6f1164942>

- [ruby/irb] fix completion test when out-of-place build

- [ruby/irb] Cache completion files to require

<https://github.com/ruby/irb/commit/612ebcb311>

- [ruby/irb] Always add input method when calling Irb.new in tests

When passes input method as nil to Context.new through Irb.new, ReidlineInputMethod.new is executed and the global internal state of Reline is rewritten, therefore other tests are failed in the Ruby repository. This commit changes to use TestInputMethod.

<https://github.com/ruby/irb/commit/010dce9210>

- [ruby/irb] Prevent the completion from crashing if rdoc is missing

There are cases where ruby is installed without rdoc and e.g. lib/irb/cmd/help.rb also handles the LoadError

Here is how to replicate the issue:

```
$ docker run -it alpine:3.13.3 sh

/ # apk add ruby ruby-irb ruby-io-console

/ # irb

irb(main):001:0> Class[TAB][TAB]
```

And you end up with something like:

```
irb(main):001:0> ClassTraceback (most recent call last):
 34: from /usr/bin/irb:23:in `<main>'
 33: from /usr/bin/irb:23:in `load'
 32: from /usr/lib/ruby/gems/2.7.0/gems/irb-1.2.6/exe/irb:11:in `<top (required)>'
 31: from /usr/lib/ruby/2.7.0/irb.rb:400:in `start'
 30: from /usr/lib/ruby/2.7.0/irb.rb:471:in `run'
 29: from /usr/lib/ruby/2.7.0/irb.rb:471:in `catch'
 28: from /usr/lib/ruby/2.7.0/irb.rb:472:in `block in run'
 27: from /usr/lib/ruby/2.7.0/irb.rb:537:in `eval_input'
 26: from /usr/lib/ruby/2.7.0/irb/ruby-lex.rb:150:in `each_top_level_statement'
 25: from /usr/lib/ruby/2.7.0/irb/ruby-lex.rb:150:in `catch'
 24: from /usr/lib/ruby/2.7.0/irb/ruby-lex.rb:151:in `block in each_top_level_statement'
 23: from /usr/lib/ruby/2.7.0/irb/ruby-lex.rb:151:in `loop'
 22: from /usr/lib/ruby/2.7.0/irb/ruby-lex.rb:154:in `block (2 levels) in each_top_level_statement'
 21: from /usr/lib/ruby/2.7.0/irb/ruby-lex.rb:182:in `lex'
 20: from /usr/lib/ruby/2.7.0/irb.rb:518:in `block in eval_input'
 19: from /usr/lib/ruby/2.7.0/irb.rb:704:in `signal_status'
 18: from /usr/lib/ruby/2.7.0/irb.rb:519:in `block (2 levels) in eval_input'
 17: from /usr/lib/ruby/2.7.0/irb/input-method.rb:294:in `gets'
 16: from /usr/lib/ruby/2.7.0/forwardable.rb:235:in `readmultiline'
 15: from /usr/lib/ruby/2.7.0/forwardable.rb:235:in `readmultiline'
 14: from /usr/lib/ruby/2.7.0/reline.rb:175:in `readmultiline'
 13: from /usr/lib/ruby/2.7.0/reline.rb:238:in `inner_readline'
 12: from /usr/lib/ruby/2.7.0/reline.rb:238:in `loop'
 11: from /usr/lib/ruby/2.7.0/reline.rb:239:in `block in inner_readline'
 10: from /usr/lib/ruby/2.7.0/reline.rb:270:in `read_io'
  9: from /usr/lib/ruby/2.7.0/reline.rb:270:in `loop'
  8: from /usr/lib/ruby/2.7.0/reline.rb:311:in `block in read_io'
  7: from /usr/lib/ruby/2.7.0/reline.rb:240:in `block (2 levels) in inner_readline'
  6: from /usr/lib/ruby/2.7.0/reline.rb:240:in `each'
  5: from /usr/lib/ruby/2.7.0/reline.rb:241:in `block (3 levels) in inner_readline'
  4: from /usr/lib/ruby/2.7.0/reline/line_editor.rb:820:in `input_key'
  3: from /usr/lib/ruby/2.7.0/reline/line_editor.rb:608:in `complete'
  2: from /usr/lib/ruby/2.7.0/irb/completion.rb:269:in `block in <module:InputCompletor>'
  1: from /usr/lib/ruby/2.7.0/rubygems/core_ext/kernel_require.rb:92:in `require'
/usr/lib/ruby/2.7.0/rubygems/core_ext/kernel_require.rb:92:in `require': cannot load such file -- rdoc (LoadError)
```

<https://github.com/ruby/irb/commit/a2d299c2ac>

- [ruby/irb] Suppress verbose messages in the parallel test

:VERBOSE flag needs to be set prior to IRB::Irb.new.

<https://github.com/ruby/irb/commit/78604682d9>

- [ruby/irb] SIGINT should raise Interrupt after IRB session

<https://github.com/ruby/irb/commit/5832cfe75b>

- [ruby/irb] Colorize __END__ as keyword

<https://github.com/ruby/irb/commit/9b84018311>

- [ruby/irb] Add show_source command

<https://github.com/ruby/irb/commit/108cb04352>

- [ruby/reline] Reset @rest_height when clear screen

<https://github.com/ruby/reline/commit/3a7019b0d5>

- [ruby/irb] process multi-line pastes as a single entity

this allows pasting leading-dot chained methods correctly:

```
class A
  def a; self; end
  def b; true; end
end
```

```
a = A.new
```

```
a
.a
.b
```

will properly return true instead of erroring on the .a line:

```
irb(main):001:1*      class A
irb(main):002:1*      def a; self; end
irb(main):003:0>      end
irb(main):004:0*
irb(main):005:0>      a = A.new
irb(main):006:0*
irb(main):007:0>      a
irb(main):008:0>      .a
irb(main):009:0>      .a
=> #<A:0x00007f984211f8e8>
```

<https://github.com/ruby/irb/commit/45aeb52575>

- [ruby/irb] Add yamatanooroti test example

<https://github.com/ruby/irb/commit/279155fcee>

- [ruby/irb] Add test for multiline paste

<https://github.com/ruby/irb/commit/e93c9cb54d>

- [ruby/irb] Evaluate each toplevel statement

<https://github.com/ruby/irb/commit/bc1b1d8bc3>

- [ruby/irb] Version 1.3.5

<https://github.com/ruby/irb/commit/22e2ddf715>

- [ruby/reline] Version 0.2.5

<https://github.com/ruby/reline/commit/22ce5651e5>

Co-authored-by: Jeremy Evans code@jeremyevans.net
 Co-authored-by: Kazuhiro NISHIYAMA zn@mbf.nifty.com
 Co-authored-by: Takashi Kokubun takashikkbn@gmail.com
 Co-authored-by: Nobuyoshi Nakada nobu@ruby-lang.org
 Co-authored-by: Aleksandar Ivanov aivanov92@gmail.com
 Co-authored-by: Koichi Sasada ko1@atdot.net
 Co-authored-by: Cody Cutrer cody@instructure.com

Revision cafa7904e7233d1a4b8ae7e738f6bd35ed67d642 - 04/03/2021 09:26 AM - aycabta (aycabta .)

Backport lib/reline, and lib/irb for 3.0.1 4th (#4349)

- [ruby/irb] Update help message for next context-mode of 4

While here, fixing tab/space issues in help message, and sync rdoc for IRB class to match the help message.

<https://github.com/ruby/irb/commit/ef8e3901cc>

- [ruby/irb] Do not continue line if last expression is an endless range

Fixes [Bug #14824]

<https://github.com/ruby/irb/commit/63414f8465>

- [ruby/irb] Add a test for not continuing when endless range at eol

<https://github.com/ruby/irb/commit/1020ac9c65>

- [ruby/irb] Make save-history extension safe for concurrent use

This makes the save-history extension check for modifications to the history file before saving it. If the history file was modified after the history was loaded and before it was saved, append only the new history lines to the history file.

This can result in more lines in the history file than SAVE_HISTORY allows. However, that will be fixed the next time irb is run and the history is saved.

Fixes [Bug #13654]

<https://github.com/ruby/irb/commit/041ef53845>

- Fix errors when XDG_CONFIG_HOME points to non-writable directory

\$HOME/.config is not writable on CI
because I think tests should not corrupt user's data.

And GitHub Actions CI sets XDG_CONFIG_HOME
since Version: 20210309.1.

https://github.com/ruby/actions/runs/2130811016?check_suite_focus=true#step:16:301

```
Errno::EACCES: Permission denied @ dir_s_mkdir - /home/runner/.config/irb
```

- Try to fix errors in TestIRB::TestHistory too

https://github.com/ruby/actions/runs/2137935523?check_suite_focus=true#step:9:562

```
1) Error:
TestIRB::TestHistory#test_history_concurrent_use:
Errno::EACCES: Permission denied @ dir_s_mkdir - /home/runner/.config/irb
/home/runner/work/actions/actions/ruby/lib/fileutils.rb:253:in `mkdir'
/home/runner/work/actions/actions/ruby/lib/fileutils.rb:253:in `fu_mkdir'
/home/runner/work/actions/actions/ruby/lib/fileutils.rb:231:in `block (2 levels) in mkdir_p'
/home/runner/work/actions/actions/ruby/lib/fileutils.rb:229:in `reverse_each'
/home/runner/work/actions/actions/ruby/lib/fileutils.rb:229:in `block in mkdir_p'
/home/runner/work/actions/actions/ruby/lib/fileutils.rb:211:in `each'
/home/runner/work/actions/actions/ruby/lib/fileutils.rb:211:in `mkdir_p'
/home/runner/work/actions/actions/ruby/lib/irb/init.rb:355:in `rc_file_generators'
/home/runner/work/actions/actions/ruby/lib/irb/init.rb:330:in `rc_file'
/home/runner/work/actions/actions/ruby/test/irb/test_history.rb:170:in `block in assert_history'
/home/runner/work/actions/actions/ruby/lib/tmpdir.rb:96:in `mktmpdir'
/home/runner/work/actions/actions/ruby/test/irb/test_history.rb:168:in `assert_history'
/home/runner/work/actions/actions/ruby/test/irb/test_history.rb:133:in `test_history_concurrent_use'
```

- [ruby/irb] Define "measure" command without forced override

<https://github.com/ruby/irb/commit/9587ba13b5>

- [ruby/irb] Add all lib files automatically

<https://github.com/ruby/irb/commit/ecc82336b7>

- [ruby/irb] Don't call Ruby 2.4+'s String#pretty_print

<https://github.com/ruby/irb/commit/89bcf107be>

- [ruby/irb] Implement ls command

<https://github.com/ruby/irb/commit/19b6c20604>

- [ruby/irb] Add whereami command

<https://github.com/ruby/irb/commit/bc822e4aac>

- [ruby/irb] Fix column overflow on ls output

<https://github.com/ruby/irb/commit/6115754623>

- [ruby/irb] Fix step's argument

cols.size was calling Integer#size, which returns 8.

Fixing a bug of <https://github.com/ruby/irb/pull/209>

<https://github.com/ruby/irb/commit/c93ae4be71>

- [ruby/irb] Deal with different screen sizes

e.g. <http://rubyci.s3.amazonaws.com/centos8/ruby-master/log/20210321T063003Z.fail.html.gz>

<https://github.com/ruby/irb/commit/ddb3472ba2>

- [ruby/irb] Have some right padding

instead of filling out an entire line

<https://github.com/ruby/irb/commit/6ac8f45f5f>

- Suppress verbose messages

Get rid of warnings in the parallel test.

unknown command: "Switch to inspect mode."

- [ruby/irb] Change ripper_lex_without_warning to a class method

<https://github.com/ruby/irb/commit/d9f8abc17e>

- [ruby/irb] Complete require and require_relative

<https://github.com/ruby/irb/commit/1c61178b4c>

- [ruby/reline] Add Reline.ungetc to control buffer

<https://github.com/ruby/reline/commit/43ac03c624>

- [ruby/reline] Reline.delete_text removes the current line in multiline

<https://github.com/ruby/reline/commit/da90c094a1>

- [ruby/reline] Support preposing and postposing for Reline.completion_proc

<https://github.com/ruby/reline/commit/1f469de90c>

- [ruby/reline] Suppress crashing when completer_{quote,word_break}_characters is empty

<https://github.com/ruby/reline/commit/c6f1164942>

- [ruby/irb] fix completion test when out-of-place build

- [ruby/irb] Cache completion files to require

<https://github.com/ruby/irb/commit/612ebcb311>

- [ruby/irb] Always add input method when calling Irb.new in tests

When passes input method as nil to Context.new through Irb.new, ReidlineInputMethod.new is executed and the global internal state of Reline is rewritten, therefore other tests are failed in the Ruby repository. This commit changes to use TestInputMethod.

<https://github.com/ruby/irb/commit/010dce9210>

- [ruby/irb] Prevent the completion from crashing if rdoc is missing

There are cases where ruby is installed without rdoc and e.g. lib/irb/cmd/help.rb also handles the LoadError

Here is how to replicate the issue:

```
$ docker run -it alpine:3.13.3 sh

/ # apk add ruby ruby-irb ruby-io-console

/ # irb

irb(main):001:0> Class[TAB][TAB]
```

And you end up with something like:

```
irb(main):001:0> ClassTraceback (most recent call last):
 34: from /usr/bin/irb:23:in `<main>'
 33: from /usr/bin/irb:23:in `load'
 32: from /usr/lib/ruby/gems/2.7.0/gems/irb-1.2.6/exe/irb:11:in `<top (required)>'
 31: from /usr/lib/ruby/2.7.0/irb.rb:400:in `start'
 30: from /usr/lib/ruby/2.7.0/irb.rb:471:in `run'
 29: from /usr/lib/ruby/2.7.0/irb.rb:471:in `catch'
 28: from /usr/lib/ruby/2.7.0/irb.rb:472:in `block in run'
 27: from /usr/lib/ruby/2.7.0/irb.rb:537:in `eval_input'
 26: from /usr/lib/ruby/2.7.0/irb/ruby-lex.rb:150:in `each_top_level_statement'
 25: from /usr/lib/ruby/2.7.0/irb/ruby-lex.rb:150:in `catch'
 24: from /usr/lib/ruby/2.7.0/irb/ruby-lex.rb:151:in `block in each_top_level_statement'
 23: from /usr/lib/ruby/2.7.0/irb/ruby-lex.rb:151:in `loop'
 22: from /usr/lib/ruby/2.7.0/irb/ruby-lex.rb:154:in `block (2 levels) in each_top_level_statement'
 21: from /usr/lib/ruby/2.7.0/irb/ruby-lex.rb:182:in `lex'
 20: from /usr/lib/ruby/2.7.0/irb.rb:518:in `block in eval_input'
 19: from /usr/lib/ruby/2.7.0/irb.rb:704:in `signal_status'
 18: from /usr/lib/ruby/2.7.0/irb.rb:519:in `block (2 levels) in eval_input'
 17: from /usr/lib/ruby/2.7.0/irb/input-method.rb:294:in `gets'
 16: from /usr/lib/ruby/2.7.0/forwardable.rb:235:in `readmultiline'
 15: from /usr/lib/ruby/2.7.0/forwardable.rb:235:in `readmultiline'
 14: from /usr/lib/ruby/2.7.0/reline.rb:175:in `readmultiline'
 13: from /usr/lib/ruby/2.7.0/reline.rb:238:in `inner_readline'
 12: from /usr/lib/ruby/2.7.0/reline.rb:238:in `loop'
 11: from /usr/lib/ruby/2.7.0/reline.rb:239:in `block in inner_readline'
 10: from /usr/lib/ruby/2.7.0/reline.rb:270:in `read_io'
  9: from /usr/lib/ruby/2.7.0/reline.rb:270:in `loop'
  8: from /usr/lib/ruby/2.7.0/reline.rb:311:in `block in read_io'
  7: from /usr/lib/ruby/2.7.0/reline.rb:240:in `block (2 levels) in inner_readline'
  6: from /usr/lib/ruby/2.7.0/reline.rb:240:in `each'
  5: from /usr/lib/ruby/2.7.0/reline.rb:241:in `block (3 levels) in inner_readline'
  4: from /usr/lib/ruby/2.7.0/reline/line_editor.rb:820:in `input_key'
  3: from /usr/lib/ruby/2.7.0/reline/line_editor.rb:608:in `complete'
  2: from /usr/lib/ruby/2.7.0/irb/completion.rb:269:in `block in <module:InputCompletor>'
  1: from /usr/lib/ruby/2.7.0/rubygems/core_ext/kernel_require.rb:92:in `require'
/usr/lib/ruby/2.7.0/rubygems/core_ext/kernel_require.rb:92:in `require': cannot load such file -- rdoc (LoadError)
```

<https://github.com/ruby/irb/commit/a2d299c2ac>

- [ruby/irb] Suppress verbose messages in the parallel test

:VERBOSE flag needs to be set prior to IRB::Irb.new.

<https://github.com/ruby/irb/commit/78604682d9>

- [ruby/irb] SIGINT should raise Interrupt after IRB session

<https://github.com/ruby/irb/commit/5832cfe75b>

- [ruby/irb] Colorize __END__ as keyword

<https://github.com/ruby/irb/commit/9b84018311>

- [ruby/irb] Add show_source command

<https://github.com/ruby/irb/commit/108cb04352>

- [ruby/reline] Reset @rest_height when clear screen

<https://github.com/ruby/reline/commit/3a7019b0d5>

- [ruby/irb] process multi-line pastes as a single entity

this allows pasting leading-dot chained methods correctly:

```
class A
  def a; self; end
  def b; true; end
end
```

```
a = A.new
```

```
a
.a
.b
```

will properly return true instead of erroring on the .a line:

```
irb(main):001:1*      class A
irb(main):002:1*      def a; self; end
irb(main):003:0>      end
irb(main):004:0*
irb(main):005:0>      a = A.new
irb(main):006:0*
irb(main):007:0>      a
irb(main):008:0>      .a
irb(main):009:0>      .a
=> #<A:0x00007f984211f8e8>
```

<https://github.com/ruby/irb/commit/45aeb52575>

- [ruby/irb] Add yamatanooroti test example

<https://github.com/ruby/irb/commit/279155fcee>

- [ruby/irb] Add test for multiline paste

<https://github.com/ruby/irb/commit/e93c9cb54d>

- [ruby/irb] Evaluate each toplevel statement

<https://github.com/ruby/irb/commit/bc1b1d8bc3>

- [ruby/irb] Version 1.3.5

<https://github.com/ruby/irb/commit/22e2ddf715>

- [ruby/reline] Version 0.2.5

<https://github.com/ruby/reline/commit/22ce5651e5>

Co-authored-by: Jeremy Evans code@jeremyevans.net

Co-authored-by: Kazuhiro NISHIYAMA zn@mbf.nifty.com

Co-authored-by: Takashi Kokubun takashikkbn@gmail.com

Co-authored-by: Nobuyoshi Nakada nobu@ruby-lang.org

Co-authored-by: Aleksandar Ivanov aivanov92@gmail.com

Co-authored-by: Koichi Sasada ko1@atdot.net

Co-authored-by: Cody Cutrer cody@instructure.com

Revision cafa7904 - 04/03/2021 09:26 AM - aycabta (aycabta .)

Backport lib/reline, and lib/irb for 3.0.1 4th (#4349)

- [ruby/irb] Update help message for next context-mode of 4

While here, fixing tab/space issues in help message, and sync rdoc for IRB class to match the help message.

<https://github.com/ruby/irb/commit/ef8e3901cc>

- [ruby/irb] Do not continue line if last expression is an endless range

Fixes [Bug #14824]

<https://github.com/ruby/irb/commit/63414f8465>

- [ruby/irb] Add a test for not continuing when endless range at eol

<https://github.com/ruby/irb/commit/1020ac9c65>

- [ruby/irb] Make save-history extension safe for concurrent use

This makes the save-history extension check for modifications to the history file before saving it. If the history file was modified after the history was loaded and before it was saved, append only the new history lines to the history file.

This can result in more lines in the history file than SAVE_HISTORY allows. However, that will be fixed the next time irb is run and the history is saved.

Fixes [Bug #13654]

<https://github.com/ruby/irb/commit/041ef53845>

- Fix errors when XDG_CONFIG_HOME points to non-writable directory

\$HOME/.config is not writable on CI
because I think tests should not corrupt user's data.

And GitHub Actions CI sets XDG_CONFIG_HOME
since Version: 20210309.1.

https://github.com/ruby/actions/runs/2130811016?check_suite_focus=true#step:16:301

```
Errno::EACCES: Permission denied @ dir_s_mkdir - /home/runner/.config/irb
```

- Try to fix errors in TestIRB::TestHistory too

https://github.com/ruby/actions/runs/2137935523?check_suite_focus=true#step:9:562

```
1) Error:
TestIRB::TestHistory#test_history_concurrent_use:
Errno::EACCES: Permission denied @ dir_s_mkdir - /home/runner/.config/irb
/home/runner/work/actions/actions/ruby/lib/fileutils.rb:253:in `mkdir'
/home/runner/work/actions/actions/ruby/lib/fileutils.rb:253:in `fu_mkdir'
/home/runner/work/actions/actions/ruby/lib/fileutils.rb:231:in `block (2 levels) in mkdir_p'
/home/runner/work/actions/actions/ruby/lib/fileutils.rb:229:in `reverse_each'
/home/runner/work/actions/actions/ruby/lib/fileutils.rb:229:in `block in mkdir_p'
/home/runner/work/actions/actions/ruby/lib/fileutils.rb:211:in `each'
/home/runner/work/actions/actions/ruby/lib/fileutils.rb:211:in `mkdir_p'
/home/runner/work/actions/actions/ruby/lib/irb/init.rb:355:in `rc_file_generators'
/home/runner/work/actions/actions/ruby/lib/irb/init.rb:330:in `rc_file'
/home/runner/work/actions/actions/ruby/test/irb/test_history.rb:170:in `block in assert_history'
/home/runner/work/actions/actions/ruby/lib/tmpdir.rb:96:in `mktmpdir'
/home/runner/work/actions/actions/ruby/test/irb/test_history.rb:168:in `assert_history'
/home/runner/work/actions/actions/ruby/test/irb/test_history.rb:133:in `test_history_concurrent_use'
```

- [ruby/irb] Define "measure" command without forced override

<https://github.com/ruby/irb/commit/9587ba13b5>

- [ruby/irb] Add all lib files automatically

<https://github.com/ruby/irb/commit/ecc82336b7>

- [ruby/irb] Don't call Ruby 2.4+'s String#pretty_print

<https://github.com/ruby/irb/commit/89bcf107be>

- [ruby/irb] Implement ls command

<https://github.com/ruby/irb/commit/19b6c20604>

- [ruby/irb] Add whereami command

<https://github.com/ruby/irb/commit/bc822e4aac>

- [ruby/irb] Fix column overflow on ls output

<https://github.com/ruby/irb/commit/6115754623>

- [ruby/irb] Fix step's argument

cols.size was calling Integer#size, which returns 8.

Fixing a bug of <https://github.com/ruby/irb/pull/209>

<https://github.com/ruby/irb/commit/c93ae4be71>

- [ruby/irb] Deal with different screen sizes

e.g. <http://rubyci.s3.amazonaws.com/centos8/ruby-master/log/20210321T063003Z.fail.html.gz>

<https://github.com/ruby/irb/commit/ddb3472ba2>

- [ruby/irb] Have some right padding

instead of filling out an entire line

<https://github.com/ruby/irb/commit/6ac8f45f5f>

- Suppress verbose messages

Get rid of warnings in the parallel test.

```
unknown command: "Switch to inspect mode."
```

- [ruby/irb] Change ripper_lex_without_warning to a class method

<https://github.com/ruby/irb/commit/d9f8abc17e>

- [ruby/irb] Complete require and require_relative

<https://github.com/ruby/irb/commit/1c61178b4c>

- [ruby/reline] Add Reline.ungetc to control buffer

<https://github.com/ruby/reline/commit/43ac03c624>

- [ruby/reline] Reline.delete_text removes the current line in multiline

<https://github.com/ruby/reline/commit/da90c094a1>

- [ruby/reline] Support preposing and postposing for Reline.completion_proc

<https://github.com/ruby/reline/commit/1f469de90c>

- [ruby/reline] Suppress crashing when completer_{quote,word_break}_characters is empty

<https://github.com/ruby/reline/commit/c6f1164942>

- [ruby/irb] fix completion test when out-of-place build

- [ruby/irb] Cache completion files to require

<https://github.com/ruby/irb/commit/612ebcb311>

- [ruby/irb] Always add input method when calling Irb.new in tests

When passes input method as nil to Context.new through Irb.new, ReidlineInputMethod.new is executed and the global internal state of Reline is rewritten, therefore other tests are failed in the Ruby repository. This commit changes to use TestInputMethod.

<https://github.com/ruby/irb/commit/010dce9210>

- [ruby/irb] Prevent the completion from crashing if rdoc is missing

There are cases where ruby is installed without rdoc and e.g. lib/irb/cmd/help.rb also handles the LoadError

Here is how to replicate the issue:

```
$ docker run -it alpine:3.13.3 sh
```

```
/ # apk add ruby ruby-irb ruby-io-console
```

```
/ # irb
```

```
irb(main):001:0> Class[TAB][TAB]
```

And you end up with something like:

```
irb(main):001:0> ClassTraceback (most recent call last):
 34: from /usr/bin/irb:23:in `<main>'
 33: from /usr/bin/irb:23:in `load'
 32: from /usr/lib/ruby/gems/2.7.0/gems/irb-1.2.6/exe/irb:11:in `<top (required)>'
 31: from /usr/lib/ruby/2.7.0/irb.rb:400:in `start'
 30: from /usr/lib/ruby/2.7.0/irb.rb:471:in `run'
 29: from /usr/lib/ruby/2.7.0/irb.rb:471:in `catch'
 28: from /usr/lib/ruby/2.7.0/irb.rb:472:in `block in run'
 27: from /usr/lib/ruby/2.7.0/irb.rb:537:in `eval_input'
 26: from /usr/lib/ruby/2.7.0/irb/ruby-lex.rb:150:in `each_top_level_statement'
 25: from /usr/lib/ruby/2.7.0/irb/ruby-lex.rb:150:in `catch'
 24: from /usr/lib/ruby/2.7.0/irb/ruby-lex.rb:151:in `block in each_top_level_statement'
 23: from /usr/lib/ruby/2.7.0/irb/ruby-lex.rb:151:in `loop'
 22: from /usr/lib/ruby/2.7.0/irb/ruby-lex.rb:154:in `block (2 levels) in each_top_level_statement'
 21: from /usr/lib/ruby/2.7.0/irb/ruby-lex.rb:182:in `lex'
 20: from /usr/lib/ruby/2.7.0/irb.rb:518:in `block in eval_input'
 19: from /usr/lib/ruby/2.7.0/irb.rb:704:in `signal_status'
 18: from /usr/lib/ruby/2.7.0/irb.rb:519:in `block (2 levels) in eval_input'
 17: from /usr/lib/ruby/2.7.0/irb/input-method.rb:294:in `gets'
 16: from /usr/lib/ruby/2.7.0/forwardable.rb:235:in `readmultiline'
 15: from /usr/lib/ruby/2.7.0/forwardable.rb:235:in `readmultiline'
 14: from /usr/lib/ruby/2.7.0/reline.rb:175:in `readmultiline'
 13: from /usr/lib/ruby/2.7.0/reline.rb:238:in `inner_readline'
 12: from /usr/lib/ruby/2.7.0/reline.rb:238:in `loop'
 11: from /usr/lib/ruby/2.7.0/reline.rb:239:in `block in inner_readline'
 10: from /usr/lib/ruby/2.7.0/reline.rb:270:in `read_io'
  9: from /usr/lib/ruby/2.7.0/reline.rb:270:in `loop'
  8: from /usr/lib/ruby/2.7.0/reline.rb:311:in `block in read_io'
  7: from /usr/lib/ruby/2.7.0/reline.rb:240:in `block (2 levels) in inner_readline'
  6: from /usr/lib/ruby/2.7.0/reline.rb:240:in `each'
  5: from /usr/lib/ruby/2.7.0/reline.rb:241:in `block (3 levels) in inner_readline'
  4: from /usr/lib/ruby/2.7.0/reline/line_editor.rb:820:in `input_key'
  3: from /usr/lib/ruby/2.7.0/reline/line_editor.rb:608:in `complete'
  2: from /usr/lib/ruby/2.7.0/irb/completion.rb:269:in `block in <module:InputCompletor>'
  1: from /usr/lib/ruby/2.7.0/rubygems/core_ext/kernel_require.rb:92:in `require'
/usr/lib/ruby/2.7.0/rubygems/core_ext/kernel_require.rb:92:in `require': cannot load such file -- rdoc (LoadError)
```

<https://github.com/ruby/irb/commit/a2d299c2ac>

- [ruby/irb] Suppress verbose messages in the parallel test

:VERBOSE flag needs to be set prior to IRB::Irb.new.

<https://github.com/ruby/irb/commit/78604682d9>

- [ruby/irb] SIGINT should raise Interrupt after IRB session

<https://github.com/ruby/irb/commit/5832cfe75b>

- [ruby/irb] Colorize __END__ as keyword

<https://github.com/ruby/irb/commit/9b84018311>

- [ruby/irb] Add show_source command

<https://github.com/ruby/irb/commit/108cb04352>

- [ruby/reline] Reset @rest_height when clear screen

<https://github.com/ruby/reline/commit/3a7019b0d5>

- [ruby/irb] process multi-line pastes as a single entity

this allows pasting leading-dot chained methods correctly:

```
class A
  def a; self; end
  def b; true; end
```

```
end

a = A.new

a
.a
.b
```

will properly return true instead of erroring on the .a line:

```
irb(main):001:1*      class A
irb(main):002:1*          def a; self; end
irb(main):003:0>      end
irb(main):004:0*
irb(main):005:0>      a = A.new
irb(main):006:0*
irb(main):007:0>      a
irb(main):008:0>      .a
irb(main):009:0>      .a
=> #<A:0x00007f984211f8e8>
```

<https://github.com/ruby/irb/commit/45aeb52575>

- [ruby/irb] Add yamatanooroti test example

<https://github.com/ruby/irb/commit/279155fcee>

- [ruby/irb] Add test for multiline paste

<https://github.com/ruby/irb/commit/e93c9cb54d>

- [ruby/irb] Evaluate each toplevel statement

<https://github.com/ruby/irb/commit/bc1b1d8bc3>

- [ruby/irb] Version 1.3.5

<https://github.com/ruby/irb/commit/22e2ddf715>

- [ruby/reline] Version 0.2.5

<https://github.com/ruby/reline/commit/22ce5651e5>

Co-authored-by: Jeremy Evans code@jeremyevans.net
Co-authored-by: Kazuhiro NISHIYAMA zn@mbf.nifty.com
Co-authored-by: Takashi Kokubun takashikkbn@gmail.com
Co-authored-by: Nobuyoshi Nakada nobu@ruby-lang.org
Co-authored-by: Aleksandar Ivanov aiivanov92@gmail.com
Co-authored-by: Koichi Sasada ko1@atdot.net
Co-authored-by: Cody Cutrer cody@instructure.com

History

#1 - 07/14/2017 09:28 AM - naruse (Yui NARUSE)

- Status changed from Open to Assigned
- Assignee set to keiju (Keiju Ishitsuka)

#2 - 03/02/2021 08:24 PM - jeremyevans0 (Jeremy Evans)

- Assignee changed from keiju (Keiju Ishitsuka) to aycabta (aycabta .)

I've submitted a pull request that makes the irb save-history extension safe for concurrent use: <https://github.com/ruby/irb/pull/193>

#3 - 03/05/2021 03:39 PM - jeremyevans0 (Jeremy Evans)

- Status changed from Assigned to Closed