

Language Update 2018 - Ruby

Kazuhiro NISHIYAMA

Learn Languages 2018
2018/08/26

自己紹介

- 西山 和広
- Ruby のコミッター
- twitter, github など: @znz

agenda

- リリース間隔 について
- 2.4.0, 2.5.0 での変更点
- 2.4.0 での大きな非互換
- 2.5.0 での主な非互換
- 2.6.0 での大きな話

LLoT 以降のリリースなど

- 2016-08-27 LLoT (前回の Language Update)
- 2016-12-24 2.4.0
- 2017-03-31 2.1 EOL
- 2017-12-25 2.5.0
- 2018-03-31 2.2 EOL
- 2018 クリスマス 2.6.0 リリース予定
- EOL は3年+年度末までが続いている
- 参考: <https://bugs.ruby-lang.org/projects/ruby/wiki/ReleaseEngineering>

2.4.0, 2.5.0 での変更点

- 速くなりました
- 機能追加されました

といった話はいつものことなので省略

2.4.0, 2.5.0 での変更点

- 標準添付ライブラリの gem 化が進んだ
 - 2.6.0 でもさらに進む予定
- `Thread#report_on_exception` でスレッドが例外終了時のバックトレース
 - 2.4.0 で追加
 - 2.5.0 でデフォルト true に

2.4.0 での大きな非互換

- Unifying Fixnum and Bignum into Integer
- 詳細は RubyKaigi 2016 の発表を参照
 - http://rubykaigi.org/2016/presentations/tanaka_ akr.html

Fixnum, Bignum とは?

- 実装の詳細で一般ユーザーにみせる必要はない
- Fixnum: 32bit 環境なら 31bit 以下の整数が効率よく扱える
- Bignum: 大きな整数もメモリーの許す限り扱える
- 普通のユーザーは区別する必要がないので Integer に統合

影響

- 利点: シンプルになる
 - 教える人にとっても勉強する人によっても良い
- 欠点: 非互換
 - 特に拡張ライブラリーに影響

非互換の例

- Sequel の DSL
 - `add_column :column, Bignum → :Bignum`
- 影響のあった拡張ライブラリ：オブジェクトをダンプ/ロードするようなものがほとんどだった
 - `ext/json, msgpack, syck, yajl, oj, ox, ruby-gnome2, etc.`
 - 2.4.0 リリース前に対応済み

Version Dependencies

- json に対する pessimistic (悲観的な) version dependency (~> 1.3)
 - 1.3 以上 2.0 未満という意味
 - 間接的にバージョン制限が入っていることが多かった
- 問題点
 - json 1.x の最新 (当時) 1.8.3 は ruby 2.4 非対応
 - 対応済みの json 2.0.x がバージョン制限で入らない

解決

- stdlib なライブラリーはバージョン依存をつけないのが推奨
<https://www.hsbt.org/diary/20160829.html>
- 結局 json 1.8.5 がリリースされて解決
<https://www.hsbt.org/diary/20170112.html>

2.5.0 での主な非互換

- rescue/else/ensure が do/end ブロック内にも直接書ける
- トップレベルの定数検索は削除

2.4 以前:

```
I0::GC #=> warning: toplevel constant GC referenced by I0::GC
```

2.5 以降:

```
I0::GC #=> NameError (uninitialized constant I0::GC)
```

backtrace の順番

2.5.0 から新しい挙動:

```
$ ruby -r time -e 'Time.parse("")'  
Traceback (most recent call last):  
  2: from -e:1:in `<main>'  
  1: from ../time.rb:370:in `parse'  
../time.rb:254:in `make_time': no time information in "" (ArgumentError)
```

状況によっては以前と同じ挙動:

```
$ ruby -r time -e 'Time.parse("")' 2>&1 | cat  
../time.rb:254:in `make_time': no time information in "" (ArgumentError)  
from ../time.rb:370:in `parse'  
from -e:1:in `<main>'
```

🔴 experimental なので今後の議論次第で変わるかも?

🔵 参考: Feature #8661 <https://bugs.ruby-lang.org/issues/>

2.6.0 の大きな変更点の一部

- safe level 廃止に向けた変更が進む
- endless range : (1..)
- JIT (Just-in-time) 対応が入る

safe level 廃止に向けた変更

- \$SAFE が 0 に戻せるようになる
- Proc#call で切り替わらなくなる

endless range

- `(1..Float::INFINITY)` や `1.step` の代わりに `(1..)`
- `when (1..)` などではかっこが必要

JIT 対応

- 実行中に gcc や clang でコンパイルする
- とりあえず仕組みが入った段階
- プログラムによっては遅くなることもある
- 高速化などは今後の課題

まとめ

- x.y.0 リリースは毎年クリスマス
- EOL は3年+年度末までが続いている
- 2.4.0, 2.5.0 での変更点
 - 2.4.0 での大きな非互換
 - json gem で起きた問題の話
- 2.5.0 での主な非互換
- 2.6.0 での大きな変更点の一部
 - safe level, endless range, JIT