
mruby と CRuby の性能比較

2015/01/27

OSS推進フォーラム アプリケーション部会

富田昌宏



Japan OSS Promotion Forum

- 軽量Ruby
- 省メモリ
- 機器組み込み
 - IOがない環境でも動作
- アプリケーションに組み込み
 - 1プロセス内に複数のmrubyインスタンス
- OS上で単体でも動作

mruby を機器組み込みではなく、Linux 上のアプリケーションに組み込む言語として使用、または CRuby の代替として使用する際の特徴を知るために、CRuby との性能を比較する。

The Ruby Benchmark Suite

- <https://github.com/acangiano/ruby-benchmark-suite.git>
- mruby で動作するようにテストを改変
(全70個中47個が動作)
- タイムアウト値を1000秒(デフォルト:300秒)
- 最大メモリ空間を 4GB に制限
- メモリ使用量は VmData を計測

mruby

IIJ mruby (github iij ブランチ 2014/12/28) をベースにモジュールを追加したもの

CRuby

ruby 2.2.0p0 (2014-12-25 revision 49005)
[x86_64-linux]

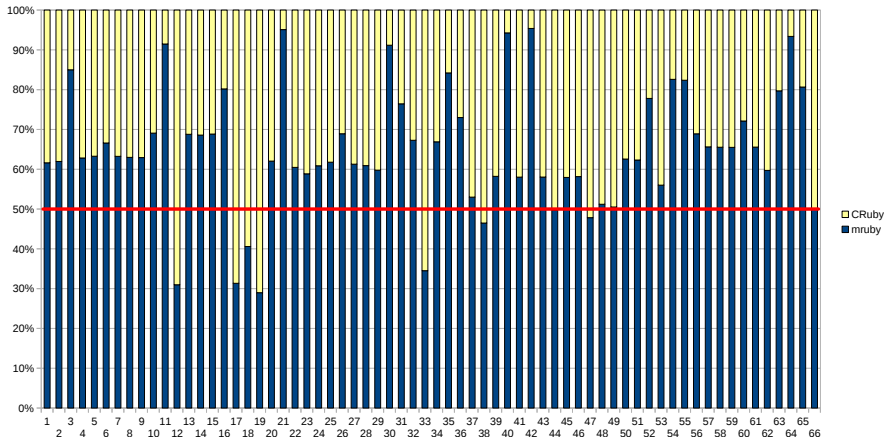
PC

ThinkPad X220 Intel(R) Core(TM) i5-2410M CPU @
2.30GHz 8GB

OS

Ubuntu 14.10 (Kernel 3.16.0-28)

測定結果



縦軸: 時間の比 / 横軸: テスト番号

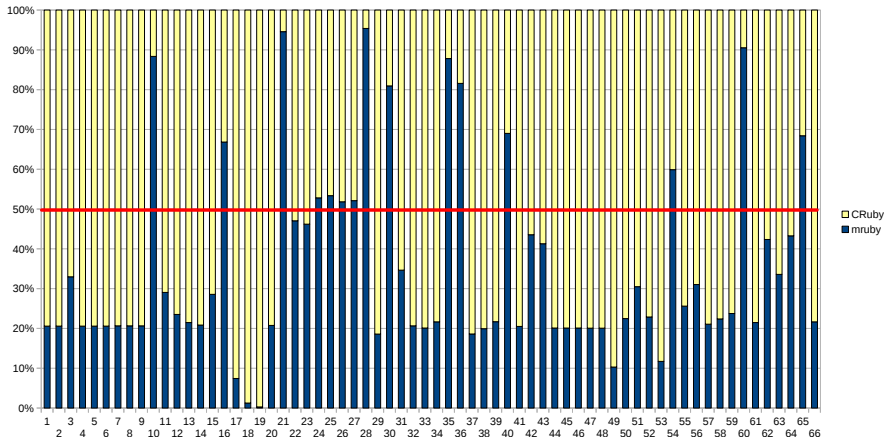
■ 速いもの

- Fiber
- eval

■ 遅いもの

- ファイル読み込み
 - CではなくRubyで書かれているため？
- 大量オブジェクト作成
 - →メモリ

メモリ消費量



縦軸: メモリ使用量の比 / 横軸: テスト番号

- 少ないもの
 - Fiber
 - TCPSocket read/write
 - 例外
- 多いもの
 - 大量のオブジェクトを作成
 - GC が追いついてないため？

mruby 対応

- `x/y` → `x.div(y)`
- `printf(io, ...)` → `io.printf(...)`
- `Dir[pattern]` → `Dir.glob(pattern)`
- 正規表現リテラル中に正規表現オブジェクトを埋め込めない
- `Numeric#zero?`, `Range#step` がない
- 等

- 各種ライブラリがない
(zlib,stringio,complex,date,pathname 等)
- 整数演算がオーバーフローし浮動小数点になる
- Thread がない
- String#tr, String#count がない
- Integer#[] がない
- 等

まとめ

- クラス/ライブラリ/メソッドが少ない
- 振る舞いが異なるものがある
- メモリ使用量は少ない
- 速度はそんなに遅くない
- GCの特性が異なる
- 性能面ではCRubyの代替としても実用的

