

Self Introduction

NAKAMURA Usaku

usa (unak)



Synergy Box

「中村」はRuby界
最強の一族

“NAKAMURA” is
the most powerful clan
in Ruby world

Three Committers



Three Committers

usa, nahi, nari



NAKAMURA vs. GOTO

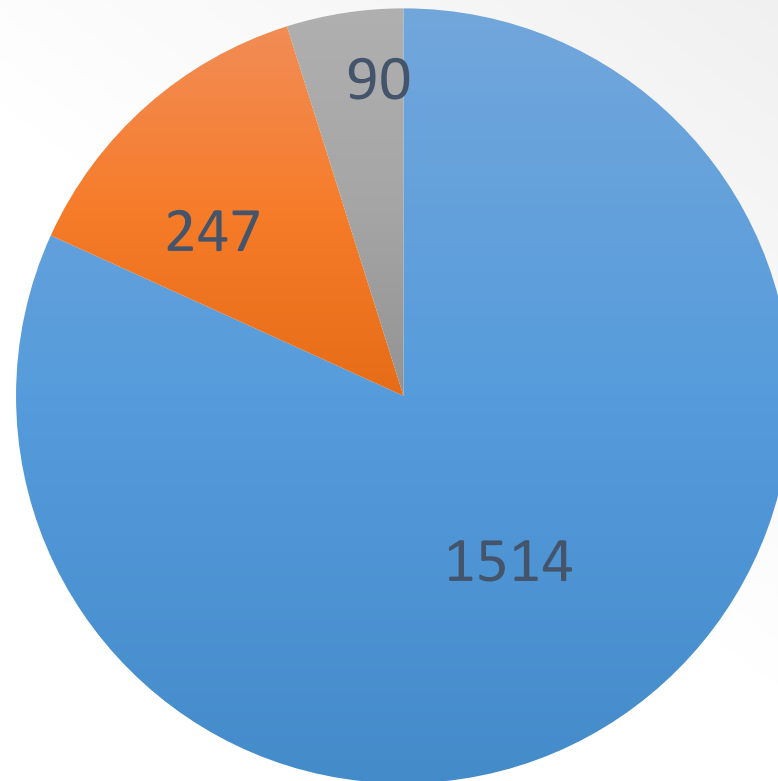
NAKAMURA : usa + nahi + nari = **1,851**

GOTO : gotoken + gotoyuzo + ngoto = **312**

※2.0.0-p0までのtrunkコミット数(kakutani調査による)

NAKAMURA vs. NAKAMURA

NAKAMURA : usa + nahi + nari = **1,851**



■ usa ■ nahi ■ nari

最強の中村一族
その中の(自称)最強

The strongest NAKAMURA
(self-professed)

Ruby 1.9.3 branch maintainer



Ruby Association



TOUA

Global Thinking

Ruby安定版保守委託事業

Windows版Ruby

メンテナ

Ruby Windows versions
maintainer

Ruby on Windows

the past, the present, and the future

Agenda

- 1.The Past : The History
- 2.The Present : What We Can Do With Ruby on Windows?
- 3.The Future : Quo Vadis?

Various Rubies on Windows

- **mswin** Win32/Win64 subsystem上。MSVCRT使用。32bitと64bitの2つ。コンパイラはVisual C++。
- **mingw** Win32/Win64 subsystem上。MSVCRT使用。32bitと64bitの2つ。コンパイラはgcc。
- **bccwin** Win32 subsystem上。Borland C++のコンパイラとCランタイム使用。死んだ。
- **cygwin** Win32 subsystem上でのUnixエミュレーション。コンパイラはgcc。
- **interix** POSIX subsystem上。コンパイラはgcc。

Windows版Rubyとは
実質的には
mswin/mingw版のこと

“Ruby Windows versions”
actually means
mswin / mingw versions

1. The History

- 1993年2月 Ruby誕生
- 1995年12月 一般向け公開(Unixのみ)
- 1996年10月 gnu-win32(後のcygwin)版登場(ebanさん)
- 1997年7月 mswin32版登場(木村さん)
- 1999年7月 この頃から小松さんがmswin32版に手を染める
- 2000年5月 ebanさんによるmingw版登場。この頃からnobuがmswin32版に手を染める
- 2000年11月 Perl/Ruby Conferenceでメンテナ募集

1. The History

- 2000年12月 usaがcommitterに
- 2001年7月 nobuがようやくcommitterに
- 2002年6月 bccwin32版登場(konishiさん)
- 2004年2月 oceanさんがcommitterに
- 2006年12月 YARVマージ
- 2007年6月 mswin64版登場(usa)
- 2010年5月 taruiさんがcommitterに
- 2010年9月 artonさんがcommitterに
- 2010年11月 luislavanaがcommitterに
- 2011年12月 shirosakiさんがcommitterに

最近のトピック

- require速度が上がった！
- mingwとmswinのCIができた！

Recent hot topics

- [開発者](#)
- [ビルド履歴](#)
- [プロジェクト相関関係](#)
- [ファイル指紋チェック](#)

ビルドキュー
なし

ビルド実行状態	
#	状態
	win7-x64
1	待機中
	win7-x86
1	待機中

All	Ruby 1.9.3	Ruby 2.0.0	Ruby Trunk			
S	W	名前 ↓	最新の成功ビルド	最新の失敗ビルド	ビルド所要時間	
		ruby-1_9_3-git	22 時間 前 - #111	—	4 分 8 秒	
		ruby-1_9_3-svn	—	—	—	
		ruby-1_9_3-x64-build	22 時間 前 - #113	1 ヶ月 3 日 前 - #100	18 分	
		ruby-1_9_3-x64-test-all	21 時間 前 - #99	5 日 20 時間 前 - #96	11 分	
		ruby-1_9_3-x86-build	22 時間 前 - #134	—	19 分	
		ruby-1_9_3-x86-test-all	21 時間 前 - #132	5 日 20 時間 前 - #129	11 分	
		ruby-2_0_0-git	22 時間 前 - #89	—	4 分 25 秒	
		ruby-2_0_0-x64-build	22 時間 前 - #85	13 日 前 - #71	28 分	
		ruby-2_0_0-x64-test-all	21 時間 前 - #78	—	12 分	
		ruby-2_0_0-x86-build	22 時間 前 - #85	13 日 前 - #71	27 分	
		ruby-2_0_0-x86-test-all	21 時間 前 - #80	—	13 分	
		ruby-trunk-git	7 時間 33 分 前 - #1392	—	8.7 秒	
		ruby-trunk-svn	—	—	—	
		ruby-trunk-x64-build	7 時間 33 分 前 - #1452	1 日 0 時間 前 - #1445	21 分	
		ruby-trunk-x64-test-all	7 時間 12 分 前 - #1268	1 日 4 時間 前 - #1260	19 分	
		ruby-trunk-x86-build	7 時間 33 分 前 - #1567	1 日 0 時間 前 - #1560	22 分	
		ruby-trunk-x86-test-all	1 ヶ月 28 日 前 - #1047	7 時間 10 分 前 - #1375	12 分	
		rubystaller-1_9_3-x64-build	20 時間 前 - #86	3 日 20 時間 前 - #84	29 分	
		rubystaller-1_9_3-x64-package	19 時間 前 - #74	3 日 18 時間 前 - #72	3 分 35 秒	
		rubystaller-1_9_3-x86-build	20 時間 前 - #124	3 日 18 時間 前 - #122	29 分	
		rubystaller-1_9_3-x86-package	20 時間 前 - #101	—	3 分 35 秒	
		rubystaller-2_0_0-x64-build	20 時間 前 - #68	3 日 18 時間 前 - #64	19 分	
		rubystaller-2_0_0-x64-package	19 時間 前 - #66	3 日 20 時間 前 - #62	3 分 32 秒	

Server	Datetime	Branch	Option	Revision	test	test-all	rubyspec	Summary	Diff
CentOS 5.6 x86_64	05-21 08:33	trunk		40868	1F0E			451W 1F0E	diff:test-all
CentOS 5.6 i386	05-21 10:03	trunk		40868	1F0E			451W 1F0E	diff:test-all
Debian 6.0.5	05-21 09:04	trunk		40868	2F2E			482W [BUG] [SEGV] 2F2E	diff:test-all
FreeBSD 9.0 amd64	05-21 09:03	trunk		40868	1F0E			524W 1F0E	diff:test-all
FreeBSD 9.1 x64	05-21 10:02	trunk		40868	1F0E			506W 1F0E	diff:test-all
OS X Mountain Lion	05-21 04:11	trunk	m32-o0	40868	1F0E			443W 1F0E	diff:src,succes...
OS X Mountain Lion	05-21 07:52	trunk	m32-o3	40868	1F0E			442W 1F0E	diff:src,succes...
OS X Mountain Lion	05-21 05:55	trunk	m64-o0	40868	1F1E			442W 1F1E	diff:src,succes...
OS X Mountain Lion	05-21 09:10	trunk	m64-o3	40868	1F0E			442W 1F0E	diff:src,succes...
Ubuntu 10.04 64-bit	05-21 09:13	trunk		40868	1F0E			451W 1F0E	diff:test-all
Ubuntu 10.04 32-bit	05-21 10:03	trunk		40868	1F0E			449W 1F0E	diff:test-all
vc10-x64	05-21 07:26	trunk		40868				433W success	no diff
CentOS 5.6 x86_64	05-21 08:59	2.0.0		40843				465W success	no diff
CentOS 5.6 i386	05-21 08:32	2.0.0		40843				464W success	no diff
Debian 6.0.5	05-21 01:46	2.0.0		40843				497W success	no diff
FreeBSD 9.0 amd64	05-21 09:39	2.0.0		40843				475W success	no diff
FreeBSD 9.1 x64	05-21 08:32	2.0.0		40843				458W success	no diff
OS X Mountain Lion	05-21 04:46	2.0.0	m32-o0	40843				459W success	diff:test-all
OS X Mountain Lion	05-21 08:17	2.0.0	m32-o3	40843				459W success	diff:test-all
OS X Mountain Lion	05-21 06:32	2.0.0	m64-o0	40843	0F1E			459W 0F1E	no diff
OS X Mountain Lion	05-21 09:37	2.0.0	m64-o3	40843				459W success	no diff
Ubuntu 10.04 64-bit	05-21 09:43	2.0.0		40843				463W success	no diff
Ubuntu 10.04 32-bit	05-21 09:39	2.0.0		40843				462W success	diff:test-all
vc10-x64	05-21 09:37	2.0.0		40843				416W success	no diff
CentOS 5.6 x86_64	05-21 09:26	1.9.3		40754				1402W success	no diff
CentOS 5.6 i386	05-21 08:59	1.9.3		40754				1401W success	no diff
Debian 6.0.5	05-21 02:23	1.9.3		40754				1698W success	no diff
FreeBSD 9.0 amd64	05-21 10:14	1.9.3		40754				1425W success	no diff
FreeBSD 9.1 x64	05-21 09:03	1.9.3		40754				1410W success	no diff
OS X Mountain Lion	05-21 05:19	1.9.3	m32-o0	40754				2880W success	no diff
OS X Mountain Lion	05-21 08:44	1.9.3	m32-o3	40754				2880W success	diff:test-all
OS X Mountain Lion	05-21 07:10	1.9.3	m64-o0	40754				1432W success	no diff
OS X Mountain Lion	05-21 10:03	1.9.3	m64-o3	40754				1432W success	no diff
Ubuntu 10.04 64-bit	05-21 10:10	1.9.3		40754				1406W success	diff:test-all,suc...
Ubuntu 10.04 32-bit	05-21 09:09	1.9.3		40754				1406W success	no diff
vc10-x64	05-21 09:00	1.9.3		40754	0F1E			1326W 0F1E	no diff
Debian 6.0.5	05-21 03:04	1.8.7		40305	1F1E	2F17E	3E7E	1550W 1NetOK 4E1E rubyspec:0F7E	diff:test-all,rub...
Debian 6.0.5	05-21 03:50	1.8.7	pth	40812	1F1E	2F17E		588W 1F1E rubyspec:2F17E	diff:test-all,rub...
Debian 6.0.5	05-21 03:33	1.8.7		40812	1F1E	2F17E		588W 1F1E rubyspec:2F17E	no diff

2. 今、Windows版Ruby
で何ができるのか？

2. What we can do with
Ruby on Windows?

Rubyって、Rails動かすの
に必要な奴でしょ?

Ruby is something that is
required for running Rails,
isn't it?

もちろん、Windowsでも
Railsは動きます!

Of course,
Rails runs on Windows!

Windowsならではの Rubyの使い方

Uses Ruby
unique to Windows

Win32OLE

Win32OLEでいろいろんな
アプリを操作

Operate various
applications by Win32OLE

るびまに連載が
あったので読んでね!

Read the series published
at Rubyist Magazine



Rubyist Magazine

Win32OLE 活用法【第 1 回】Win32OLE ことはじめ

検索

B! 226

ツイート < 18

いいね! < 2

更新日時:2006/06/22 22:46:29

書いた人:cuzic

はじめに

受信ボックスにたまった Excel ファイルを添付したメール。Excel の表にたがって、DB から検索して、特定のセルにを入力を繰り返す面倒な作業。気が滅入ります。

今まで、皆さんはこういう面倒な作業を効率化するためにどんなアプローチをしてきましたか？たとえば、MS Office には Visual Basic for Application (VBA) を使うことで自動的に処理させることができます。だけど、VB はあまりよく知らないし、使い慣れた Ruby で自動化していきたいって思ったことがありますよね。

こんなときに活躍するのが Win32OLE です。この連載では Win32OLE という Ruby の拡張ライブラリを使ったプログラムについて紹介していきます。Win32OLE という名前からわかるとおり、Microsoft Windows (以下、Windows) でしか動作しません。そのため、本連載は Windows を利用している Rubyist を対象としています。

Win32OLE は、COM とか ActiveX などと呼ばれたりする技術を扱うためのライブラリです。この記事の中では OLE/COM/ActiveX といった用語をあまり区別せずに COM という言葉でまとめておきます。Windows アプリケーションは一般にいくつかの部品を組み合わせてできています。この部品化する技術の中に Microsoft 社が開発した技術に Component Object Model (COM) があります。

COM には言語に非依存であるという特徴があります。つまり、COM で作られた部品であれば C++ でも VB でも Delphi でも、そして Ruby でもプログラムの中で扱うことができます。

こう言ってもピンとこないかもしれません。具体的に言うと、Ruby で書いたプログラムで Internet Explorer を制御した

0042号 (2013-05)

巻頭言

Rubyist Hotlinks ト部さん

Cucumber から Tumip へ

クックパッド 2.0.0 対応

桐島、Rubyやめるってよ

Ruby 20 周年パーティー

Regionalレポート福岡01

Regionalレポート東京10

Regionalレポートくまみ01

Regionalレポート大江戸03

るびま移行後記

Rubyの歩き方

RubyEventCheck

編集後記

バックナンバー

各号目次

分野別目次

2.0.0 Special (En)

0041号 (2013-02)

0040号 (2012-11)

0039号 (2012-09)

RubyKaJaのご紹介

0038号 (2012-06)

0037号 (2012-02)

0036号 (2011-11)

0035号 (2011-09)

本気出せば
Rubyだけで
GUIアプリも書ける

You can write GUI applications
with Ruby alone,
if you give it your all

Windowsでこそ
Rubyは活きる

Windows is where
Ruby really shines

3. Windows版Rubyの 今後

3. Quo Vadis?

目指すもの

Goals

目指さないもの

- Unixエミュレーション
- Windows独自機能の網羅
- CRuby本体からの分離独立

Non-goals

Unix

エミュレーション?

Unix Emulation?

◆不定期連載『なるほどUnixプロセス』で学ぶWindows版Rubyの基礎～第11章 forkとか死ねばいいのに

最初に明言しておく。好んでfork(2)を使う奴とは友達にはなれない。

fork(2)がどんなものなのかはそれこそ『なるほどUnixプロセス』を読んでもらうとして、ぶっちゃけお前らプロセスのコピーなんかほしいの?というのが正直なところである。

いや、もちろん、便利な場面もあるだろう。

しかし、処理を分散したいならわざわざ別プロセスを生成しなくてもスレッドを使えばいいではないか。

fork(2)に固執するのは単なる老害かUnix厨である。消えてしまえ!

で、えーと、結論としては、Windows版RubyではKernel#forkはNot ImplementedError例外を投げる。

スレッドかKernel#spawnとかを使ってくれたまえ。

◆不定期連載『なるほどUnixプロセス』で学ぶWindows版Rubyの基礎～第12章 全てのプロセスは孤児である

[この前書いた](#)が、基本的にはWindowsにおいては全てのプロセスは生まれながらに孤児である、とみなしてよい。

あれ、これくらいしかこの章について言うことないな。

えーと、『なるほどUnixプロセス』の本章のサンプルコードは、Windowsではこんな感じで(1)どうだろうか?

```
r, w = IO.pipe
w.write <<END_OF_SCRIPT
5.times do
  sleep 1
  puts "I'm an orphan!"
end
```

Windows 独自の
機能の網羅？

Implement all Windows-
specific features?

CRuby本体から
の分離独立？

Independent from CRuby?

これからやりたいこと

- コマンドラインUnicode化
- その他Unicode化漏れ部分
- Cancelable I/O (revised)

Want to do

コマンドライン Unicode化

Unicodizing commandline

C:\ コマンドプロンプト

```
C:\>ruby -e "p ARGV" 森鷗外  
["森?外"]
```

```
C:\>■
```

その他Unicode化 漏れ部分

Other Unicode necessities

C:\ コマンドプロンプト

```
C:¥>nkf -Ws test.rb
```

```
# coding: utf-8
```

```
`にほんご.exe`
```

```
C:¥>ruby test.rb
```

```
test.rb:2:in ``: No such file or directory - `にほんご.exe` (Errno::ENOENT)  
    from test.rb:2:in `<main>'
```

```
C:¥>■
```


Cancelable I/O (revised)

```
require "socket"
server = TCPServer.new(12345)
th = Thread.new(server.accept) do |sock|
  data = sock.recv
  # ...
end

# ...
if something_occurred
  th.kill
end
```

ご清聴ありがとうございました。
ございました。

Thank you for your attention.