

Fluentd の開発支援の話

Fluentd の Windows 版の機能に関わる開発支援の裏話

Hiroshi Hatake

株式会社クリアコード

Fluentd meetup in Matsue

自己紹介

- Hiroshi Hatake



- Twitter: @cosmo__
 - GitHub: @cosmo0920
- 株式会社クリアコード
- OSS サポート (開発・導入支援・時間制サポート etc.)
をしています。¹

¹<https://www.clear-code.com/services/floss/development.html>

- ① はじめに
- ② Fluentd の開発支援の話
- ③ Fluentd v0.14 の新機能のおさらい
- ④ Fluentd の Windows サポート
- ⑤ Windows での Ruby のエコシステム
- ⑥ win32-api gem のサポート
- ⑦ windows-pr gem の Fluentd に関わる Issue の解決
- ⑧ まとめ

Fluentd の開発支援の話

開発支援でやってきた内容

- Fluentd 0.12.16 の secret parameter が入った前後から開発支援をしています
- メモリダンプの機能
- Parser/Formatter のテストドライバ
- built-in のプラグインの v0.14 の API への移行
- AppVeyor の導入のお手伝い
- メンテナンスが活発でない gem を引き取ってメンテナンス
- プラグインへ各種 PR etc.

Fluentd の開発支援の話

開発支援でやってきた内容

- Fluentd 0.12.16 の secret parameter が入った前後から開発支援をしています
- メモリダンプの機能
- Parser/Formatter のテストドライバ
- built-in のプラグインの v0.14 の API への移行
- AppVeyor の導入のお手伝い
- メンテナンスが活発でない gem を引き取ってメンテナンス
- プラグインへ各種 PR etc.

Fluentd v0.14 の開発支援をしていく中で、
Fluentd の Windows 向けの機能で依存して
いる gem のメンテナンスを引き取った話
をします。

Fluentd v0.14 の新機能³のおさらい

- Windows サポート
- 高精度な時刻サポート
- 新しいプラグイン API
- router の使用の強制²

²Engine.emit がバグ扱いになりました

³<http://www.fluentd.org/blog/fluentd-v0.14.0-has-been-released>

Fluentd v0.14 の新機能³のおさらい

- Windows サポート
- 高精度な時刻サポート
- 新しいプラグイン API
- router の使用の強制²

²Engine.emit がバグ扱いになりました

³<http://www.fluentd.org/blog/fluentd-v0.14.0-has-been-released>

つまり、Fluentd v0.14 の開発では Windows も考慮した開発が必要。

つまり、Fluentd v0.14 の開発では **Windows** も考慮した開発が必要。

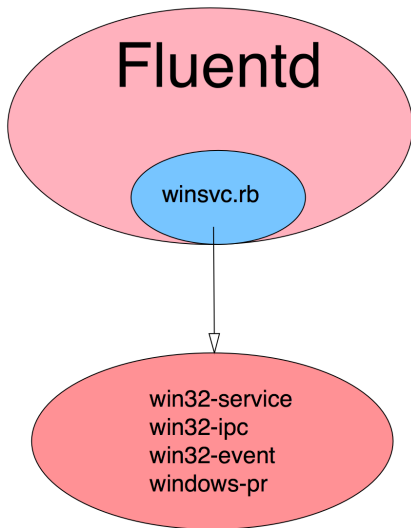
Fluentd の Windows 版で増えている依存関係 (抜粋)

```
if /mswin|mingw/ =~ RUBY_PLATFORM
  gem.add_runtime_dependency("win32-service", ["~>0.8.3"])
  gem.add_runtime_dependency("win32-ipc", ["~>0.6.1"])
  gem.add_runtime_dependency("win32-event", ["~>0.6.1"])
  gem.add_runtime_dependency("windows-pr", ["~>1.2.5"])
end
```

Windows 版で増えている依存関係

- win32-service
- win32-ipc
- win32-event
- windows-pr
 - windows-api
 - win32-api

今回話す話す内容の gem と Fluentd の関係図



Windows サービスに関わる箇所 (winsvc) が依存しています。

今回話す話す内容に関わる gem たち

- windows-pr
- windows-api
- win32-api

今回話す話す内容に関わる gem たち

- windows-pr
- windows-api
- win32-api C 拡張を含む gem

C 拡張とは？

C 拡張で何かできるの？

Ruby には C extension という C により Ruby を拡張できる機能があります。

これにより、C のライブラリの機能を Ruby に取り込むことができます。

C 拡張とは？

C 拡張で何かできるの？

Ruby には C extension という C により Ruby を拡張できる機能があります。

これにより、C のライブラリの機能を Ruby に取り込むことができます。

代表例：

C 拡張とは？

C 拡張で何かできるの？

Ruby には C extension という C により Ruby を拡張できる機能があります。

これにより、C のライブラリの機能を Ruby に取り込むことができます。

代表例：

- Windows の COM の機能をバインドした Win32OLE

C 拡張とは？

C 拡張で何かできるの？

Ruby には C extension という C により Ruby を拡張できる機能があります。

これにより、C のライブラリの機能を Ruby に取り込むことができます。

代表例：

- Windows の COM の機能をバインドした Win32OLE
- Groonga の C ライブラリをバインドした Rroonga

C 拡張とは？

C 拡張で何かできるの？

Ruby には C extension という C により Ruby を拡張できる機能があります。

これにより、C のライブラリの機能を Ruby に取り込むことができます。

代表例：

- Windows の COM の機能をバインドした Win32OLE
- Groonga の C ライブラリをバインドした Rroonga
- GTK+ のライブラリをバインドした Ruby-GNOME2

C 拡張とは？

C 拡張を含む gem の注意点

- gem のユーザーは依存しているライブラリをシステムにインストールする必要があります
- gem のユーザーは C のソースをビルドするための開発環境をシステムにインストールする必要があります

C 拡張とは？

C 拡張を含む gem の注意点

- gem のユーザーは依存しているライブラリをシステムにインストールする必要があります
 - 実は `dll` や `dylib`、`so` だけでも Ruby の C 拡張から機能が呼べることであれば使う分には大丈夫です。
- gem のユーザーは C のソースをビルドするための開発環境をシステムにインストールする必要があります

Windows での Ruby のエコシステム - fat gem とは

fat gem とは

- **fat gem** という gem の中に Ruby の C 拡張のバイナリをパッケージングできるしくみがあります。
- Fluentd でも使っている cool.io⁴ でもこの仕組みを使用しています。

⁴<https://github.com/tarcieri/cool.io>

Ruby の C 拡張のバイナリを gem にパッケージング

利点・欠点

- 利点
 - C 拡張を予め入れておくことで、Windows で Ruby を使うユーザーが C 拡張をビルドしなくてもよくなります
- 欠点
 - 新しい Ruby が出たらそれ用の gem をリリースしなければならない
 - 開発者がパッケージングする手間が増える
 - クロスコンパイルするのに一手間

Ruby の C 拡張のバイナリを gem にパッケージング

それでも fat gem を提供する理由

- ユーザーからすると fat gem が提供されていた方が嬉しい
- Windows ユーザーはあまり開発環境を構築していない

win32-api とは

元は djberg96 氏⁵ 作。

Win32API を Ruby から呼び出せるラッパー。

Ruby 本体の API にはないコールバックサポートがあります。
`#include < windows.h > # WIN32API` のおまじない

⁵<https://github.com/djberg96>

win32-api の Ruby 2.3 サポート

Ruby 2.3 がサポートされていなかったので Issue を上げました

Support precompiled binaries for Ruby 2.3:

<https://github.com/djberg96/win32-api/issues/16>

win32-api の Ruby 2.3 サポート

@djberg98 “@cosmo0920 I don't suppose you would be interested in taking over this project, would you?”⁶

⁶<https://github.com/djberg96/win32-api/issues/16#issuecomment-212021651>

つまりどういうこと？

真意がよくわからないので聞いてみましょう

- @cosmo0920(me) “What do you want to do for me? Just building universal gem? Or, entirely taking over this project?”

- @cosmo0920(me) “What do you want to do for me? Just building universal gem? Or, entirely taking over this project?”
- @djberg96 “@cosmo0920 Completely taking over project.”

win32-api の Ruby 2.3 サポート



djberg96 commented on 27 Apr

@cosmo0920 Completely taking over project.

win32-api の Ruby 2.3 サポート



djberg96 commented on 27 Apr

@cosmo0920 Completely taking over project.

訳：完全に引き継いでください。

win32-api の Ruby 2.3 サポート



djberg96 commented on 27 Apr

@cosmo0920 Completely taking over project.

訳：完全に引き継いでください。

win32-api の Ruby 2.3 サポート

実はメンテナを探していた

= Maintainer Wanted! Since I no longer use this project, I would like to turn it over to someone who has the skill, time and desire to keep it going.⁷

⁷<https://github.com/cosmo0920/win32-api/blob/eef0b35dd095f43cac0d48824a782ecbb31a25d6/README#L118>

win32-api の Ruby 2.3 サポート

実はメンテナを探していた

= Maintainer Wanted! Since I no longer use this project, I would like to turn it over to someone who has the skill, time and desire to keep it going.⁷

訳: メンテナ求む! このプロジェクトはもう使っていないので、技術があり、時間と続けていく心意気のある誰かに譲りたい。

⁷<https://github.com/cosmo0920/win32-api/blob/eef0b35dd095f43cac0d48824a782ecbb31a25d6/README#L118>

win32-api プロジェクトの master を引き継ぎました。

win32-api プロジェクトの master を引き継ぎました。

windows-api, windows-pr プロジェクトの master も合わせて引き継ぎました。

win32-api の Ruby 2.3 サポート

Ruby 2.3 に対応させる⁸ 作業と、AppVeyor⁹ の導入¹⁰ の作業を行いました。

Support Ruby 2.3 #18

Merged cosmo0920 merged 2 commits into master from support-ruby

Conversation 2 Commits 2 Files changed 1



cosmo0920 commented on 5 Jun

This is the revised version of #17

cosmo0920 added some commits on 27 Apr

- Support Ruby 2.3 build
- Drop Ruby 1.8.7 and 1.9.3

cosmo0920 added some commits on 6 Jun

- remove needless requires
- Fix Ruby 2.0.0 paths with RubyInstaller's default installation paths
- Add appveyor.yml
- Use bundle

2c18f84

dc1879d

3792cc0

52b1340

All checks have passed

2 successful checks

cosmo0920 comm

Rebased and I've g
Can I merge this?

continuous-integration/appveyor/branch — AppV...

Details

continuous-integration/appveyor/pr — AppVeyor ...

Details

⁸<https://github.com/cosmo0920/win32-api/pull/18>

⁹<https://www.appveyor.com/>

¹⁰<https://github.com/cosmo0920/win32-api/pull/20>

1.6.0 として Ruby 2.3 のサポートと、Ruby 1.8.7 と 1.9.0 のサポートを切ったバージョン 1.6.0 をリリース済みです。¹¹

¹¹<https://rubygems.org/gems/win32-api/versions/1.6.0-universal-mingw32>

windows-api gem のサポート

windows-api gem

AppVeyor の導入をしました。¹²

The screenshot displays a GitHub pull request interface. At the top, a comment from user **cosmo0920** dated 16 Jun states: "CI target Ruby version is 2.0.0 or later. Because win32-api 1.6.0 does not contain 1.8.7 and 1.9.3 binaries." Below this, a commit summary shows **cosmo0920** added some commits on 16 Jun, including "Add appveyor.yml" and "Add rake i...". A green badge indicates "All checks have passed" with "2 successful checks". Two check runs are listed: "continuous-integration/appveyor/branch" and "continuous-integration/appveyor/pr", both with green checkmarks and "Details" links. A comment from **djberg96** with a thumbs-up icon is visible. Further down, a merge commit by **cosmo0920** on 17 Jun is shown, merging commit **5e85f96** into the **master** branch, with a note "2 checks passed" and links for "View details" and "Revert". At the bottom, a message indicates **cosmo0920** deleted the **appveyor** branch on 17 Jun, with a "Restore branch" link.

cosmo0920 commented on 16 Jun

Owner + 🗨️ ✎

CI target Ruby version is 2.0.0 or later.
Because win32-api 1.6.0 does not contain 1.8.7 and 1.9.3 binaries.

cosmo0920 added some commits on 16 Jun

- Add appveyor.yml 08f991e
- Add rake i... 55995f8

All checks have passed
2 successful checks

- continuous-integration/appveyor/branch — AppV... Details
- continuous-integration/appveyor/pr — AppVeyor ... Details

djberg96 commen

👍

cosmo0920 merged commit 5e85f96 into master on 17 Jun
2 checks passed View details Revert

cosmo0920 deleted the appveyor branch on 17 Jun Restore branch

¹²<https://github.com/cosmo0920/windows-api/pull/6>

windows-pr gem のサポート

メンテナンスを引き継いでからすぐに Issue を解決することとなりました。¹³

対応する Fluentd 側の Issue は

<https://github.com/fluent/fluentd/issues/920>

¹³<https://github.com/cosmo0920/windows-pr/issues/13>

windows-pr gem のサポート

in_tail が Windows 上で問題を起こしていた

in_tail で監視しているファイルが削除されると、
*undefinedmethod 'pe =' for # < Fluent :: TailInput ::
TailWatcher :: NullIOHandler :: 0x00000000XXXXXXXX > ''*
というエラーが起きていました。

何が起きていたのか

- INVALID_HANDLE_VALUE が 32bit と 64bit では異なる定数として扱わなければなりません。しかし、0xFFFFFFFF がハードコートされてしまっていました。
- 64bit 環境では INVALID_HANDLE_VALUE が 0xFFFFFFFFFFFFFFFF となるように修正しました。¹⁴
15

¹⁴<https://github.com/cosmo0920/windows-pr/pull/15>

¹⁵ただし、この修正は CRuby に対しては有効で、JRuby に対してはバグを含んでいます。

windows-pr gem のサポート

Use 64bit adopted constant #15

 **Merged** cosmo0920 merged 1 commit into `master` from `use-64bit-adopted-constant` on 22 Jun

 Conversation **7**

 Commits **1**

 Files changed **2**




cosmo0920 commented on 20 Jun

Owner



Based on
[#13 \(comment\)](#)
and
[#13 \(comment\)](#)
patches.

  Use 64bit adopted constant ...

✓ debaba6

windows-pr gem のサポート

無事、治ったとのこと。

cosmo0920 commented on 22 Jun



I've released windows-pr 1.2.5.
Please confirm this issue has been fixed.

tako774 commented on 22 Jun • edited



I updated window-pr and confirmed this issue didn't happen with both ruby2.2.4 x86/x64 for the scenario which is 2 times loop of make file -> delete file.

Thanks for your great job! 🍷🐱🍷



1

INVALID_FILE_HANDLE の 64bit Windows 環境での定数の不具合を修正したバージョン 1.2.5 をリリース済みです。¹⁶

¹⁶<https://rubygems.org/gems/windows-pr/versions/1.2.5>

まとめ

Fluentd の Windows 版の機能で依存している gem のメンテナンスを引き取った話をしました。

その際に、単に引き取るだけでなく、よりメンテナンスがしやすい方向にしていく変更を入れました。

Windows で Ruby を使う際には、C 拡張や Windows に関わる Ruby や gem の問題に対応しているメンテナの存在を思い出してあげてください。