

CとRubyとその間

須藤功平

株式会社クリアコード
2009/07/18

話すこと

2.times do

- ✓ CとRubyの得手不得手
- ✓ よいとこどりをするには

end

得手不得手

✓ Ruby

✓ 得意: 直感的・柔軟

✓ 苦手: 高速な動作

✓ C

✓ 得意: 高速な動作

✓ 苦手: 直感的・柔軟

補完

- ✓ RubyとCは似ていない
- ✓ だから仲良くできる

パターン

- ✓ Rubyの苦手をカバー
- ✓ Rubyの得意を活かす

苦手をカバー

- ✓ Rubyの**苦手**をカバー
- ✓ Rubyの得意を活かす

Rubyの苦手をカバー

苦手: **高速な動作**

✓ API: Ruby

✓ 実装: C

→ **拡張ライブラリ**

ラングバプロジェクトはカラムストア機能も備える高速・高機能な全文検索エンジンgroongaの機能をRubyから利用するためのライブラリを提供するプロジェクトです。groongaの機能をRubyらしい読み書きしやすい構文で利用できることが利点です。

例: 全文検索



groonga

- ✓ **高速な**全文検索エンジン
- ✓ **高速な**カラムストア
- ✓ 得意な用途
 - ✓ 常時更新・常時検索
- ✓ **C**で実装

柔軟で高速な動作

✓ Ruby

✓ 得意: 直感的・柔軟

✓ ~~苦手: 高速な動作~~

✓ C

✓ 得意: **高速な動作**

✓ ~~苦手: 直感的・柔軟~~



役割分担

ActiveGroonga



- ActiveRecord風API



Ruby/groonga



- RubyらしいAPI
 - 速度は殺さない
- 素材を活かす

groonga



- 高速な全文検索エンジン
- 高速なカラムストア

祝0.0.3

リリース

利用場所
(予定)



総記事数：61090 | 総レス数：575091 | 総お絵カキコ数：44844 | 総ピコカキコ数：2004

ニコニコ大百科は、(有)未来検索ブラジルが運営するあらゆる言葉についての記事を読覧・編集したり、コメントをしたりするサイトです。

- [06/30] [仕様変更]お絵カキコのキャンパス回転の中心点がキャンパス中央に移動しました。
- [06/30] [機能追加]ピコカキコエディタでundo/redoができるようになり、フォントが選べるようになりました。

[新着お絵カキコ](#) / [新着ピコカキコ](#) / [新着レス](#) / [新着単語](#)・[動画](#)・[商品記事](#) / [新着ユーザ](#)・[コミュニティ記事](#)

ニコニコ大百科(仮)

★愛されてピコカキコ2,000を突破!★

ニコニコ大百科は、様々な単語や動画について記事を編集したり、掲示板にコメントできるサイトです。

お絵カキコより [今週の扉絵](#) →
はくはく ◆Ohi4gelJXYについて語るスレ

- 単語記事 : [プレミアム会員](#)のみ編集可能
- 動画記事 : 動画をアップロードした会員か
- 単語記事 : [プレミアム会員](#)のみ編集可能
- 商品記事 : [プレミアム会員](#)のみ編集可能
- 郵便記事 : [プレミアム会員](#)のみ編集可能
- 東証記事 : [プレミアム会員](#)のみ編集可能



buzztter

Twitter のイマを切り取った一☆



News: buzztter IRC bot が稼働中です: #buzztter on irc.freenode.net

たいやき OR Ruby

検索

こんなクエリはいかがですか? [投票率](#) [当確](#) [開票速報](#)

buzztter

投票率 当確 都議選のハッポツ議席 スナイプ 民主が開票速報 自民が共産 ハーマイオニー 公明 マルフォイ 民主ルーピソアズ カバン 都民ハリー 圧勝 自民党 当選 投票しの選挙 選挙区 フォイ 票 政党 都議会 #tdf 政治の結果 帰宅 お風呂 風呂 支持 党 区 候補 政権 ただいまことになっ ビール 寝る 速報 t 疲れ 麻生 高専 明日のかわいい 見てる たんだろ 22 ねえ 第一 負けたのは 先生 結局 酒 これで 帰って dq9 たw 国民 か・・・ まあ でした 飲ん 部屋

たいやき OR Ruby 自動更新: stop



DONATE pledgie.com
\$80.00 Raised!

12星座MMORPG
Lucent Heart

12星座MMORPG
Lucent Heart

\$80.00 Raised!

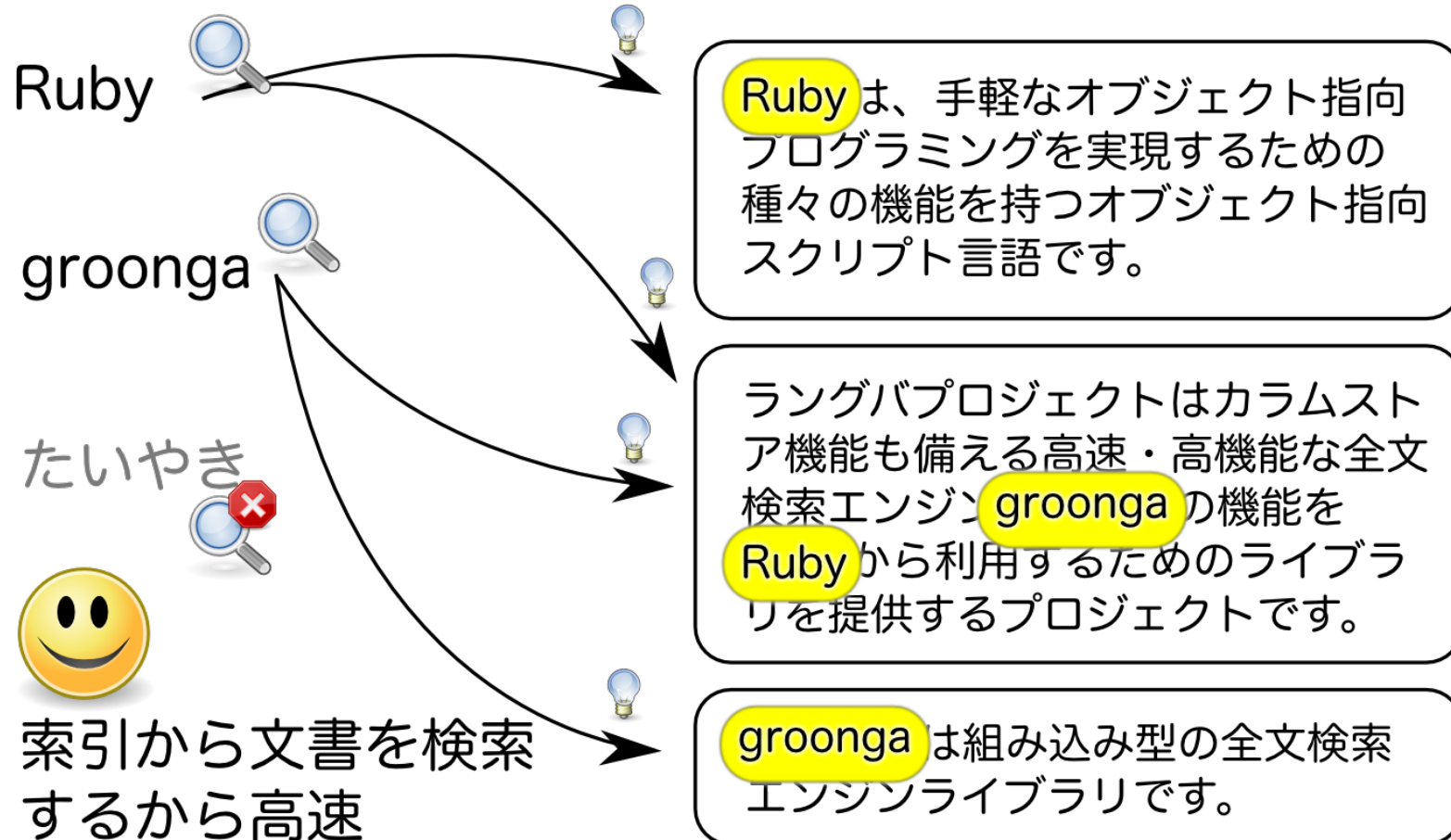
groonga: 全文検索

- ✓ 高速な**全文検索**エンジン
- ✓ 高速なカラムストア
- ✓ 得意な用途
 - ✓ 常時更新・常時検索

全文検索

索引

文書





Ruby

逆引きレシピ

島田浩二 / 設樂洋爾 / 村田賢太 著
前田智樹 / 谷口文威

すぐに美味しい
サンプル&
テクニック
232

索引から検索

★★★
プロが選んだ三ツ星レシピ!!
広大なRubyの世界でもう迷わない!
開発現場で今日から使える
とっておきのノウハウ集

“やりたいこと”からすぐ引ける! アプリケーション開発時に
即役立つ機能・ライブラリ・テクニックを丁寧に解説。



groonga: カラムストア

- ✓ 高速な全文検索エンジン
- ✓ 高速なカラムストア
- ✓ 得意な用途
 - ✓ 常時更新・常時検索

カラムストア

💡 データを縦方向に扱う

- 固定長データ
- 高速アクセス

キー

ID

C	1
Ruby	2
Scheme	3



特長
カラム

文字列

高速な動作

直感的な記述

シンプル



誕生日
カラム

数値

1972

1995

197X



索引もカラム



複数の索引を利用可

- 特性が異なる
- 使い分ける

単語

ID

C	1
Ruby	2
Ru	3



用途に応じた
柔軟なシステム

N-gram
カラム

転置索引
カラム

...
...
...

...
...
...

Ruby + groonga

- ✓ Ruby/groonga
 - ✓ RubyらしいAPI
 - ✓ groongaのよさは殺さない
- ✓ ActiveGroonga
 - ✓ ActiveRecord風API

Ruby/groonga

```
users = Groonga::Array.create
users.define_column(:name, "Text")
ko1 = users.create(:name => "ko1")
ko1["name"] # => "ko1"
```

ActiveGroonga

```
class User < ActiveGroonga::Base
end
ko1 = User.create(:name => "ko1")
ko1.name # => "ko1"
```

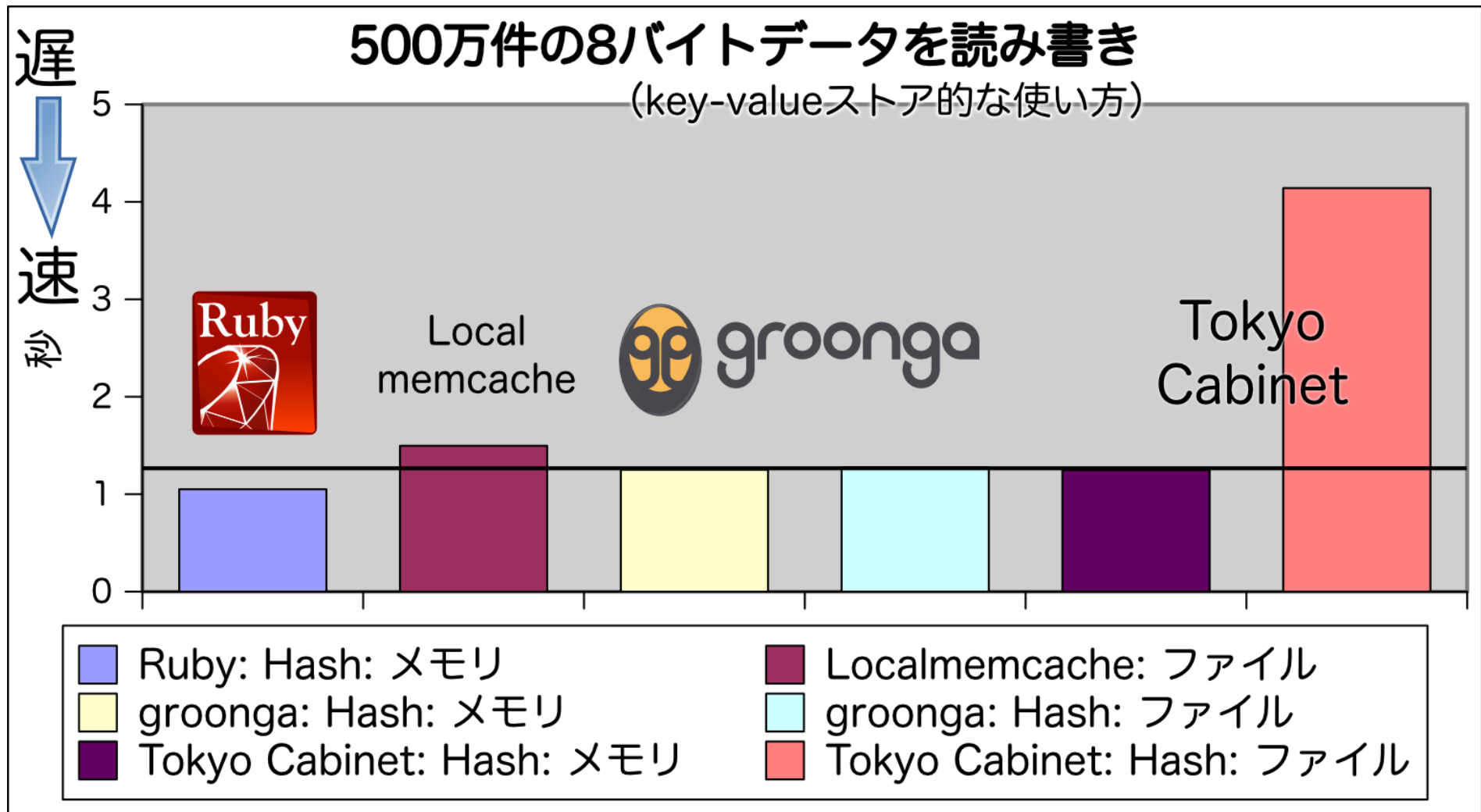
groonga + Rails

✓ フレームワーク対応

```
config.frameworks -= [:active_record]  
config.frameworks += [:active_groonga]
```

✓ マイグレーション対応

参考: ベンチマーク



まとめ: 苦手をカバー

- ✓ 実装: **高速**なgroonga
- ✓ API: より使いやすく
 - ✓ Ruby/groonga: RubyらしいAPI
 - ✓ ActiveGroonga: RailsらしいAPI

拡張ライブラリで
高速で使いやすいライブラリ

話していること

```
2.times do |i|
```

```
✓ pi # => 1
```

```
✓ CとRubyの得手不得手
```

```
✓ よいところどりをするには
```

```
end
```

得手不得手

✓ Ruby

✓ 得意: 直感的・柔軟

✓ 苦手: 高速な動作

✓ C

✓ 得意: 高速な動作

✓ 苦手: 直感的・柔軟

得意を活かす

- ✓ Rubyの苦手をカバー
- ✓ Rubyの**得意を活かす**

Rubyの得意を活かす

得意: **直感的・柔軟**

✓ インターフェイス: Ruby

✓ コア: C

→ **Rubyインタプリタ埋込**

例：迷惑メーラー対策



milter manager

祝 1.2.0

リリース

filter manager

- ✓ メール毎に異なる対策を適用
 - ✓ 誤検出を減らす
 - ✓ 効率アップ
 - ✓ ユーザの意志を尊重
- ✓ 大量のアクセスを処理
- ✓ **C**で実装

必要な機能

- ✓ **直感的な**設定方法
- ✓ **柔軟な**動作変更
 - ✓ 怪しそうなメールには対策を適用
 - ✓ 宛先毎に対策を変える
 - ✓ 正規表現・RDB/LDAPアクセス

柔軟で高速な動作

✓ Ruby

✓ **得意: 直感的・柔軟**

✓ ~~苦手: 高速な動作~~

✓ C

✓ **得意: 高速な動作**

✓ ~~苦手: 直感的・柔軟~~

役割分担

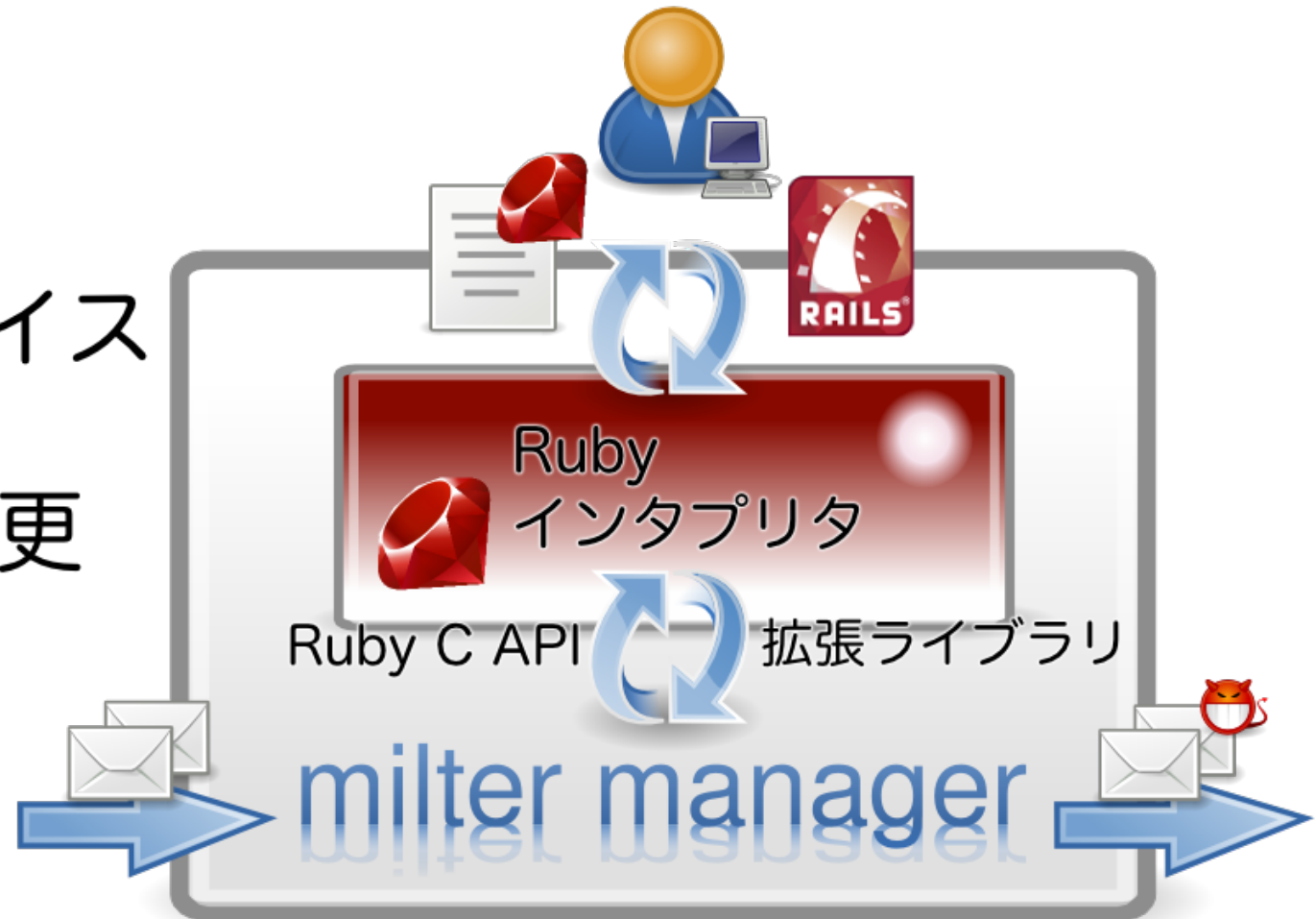


管理者との
インターフェイス



動的な動作変更

高速な
メール処理



The Ruby Logo: Copyright (c) 2006, Yukihiro Matsumoto

直感的な設定ファイル

分類.キー = 値

```
manager.daemon = true
```

```
security.effective_user = "mail"
```

```
controller.unix_socket_mode = 0660
```

直感的な設定方法

- ✓ 設定なしで動く
- ✓ 適切なデフォルト
 - ✓ システムの状況から **自動検出**
- ✓ 簡単なものはWebのUI

柔軟な動作変更

```
def stop_apply?(address)
  return false if /@test\.example\.com\z/ =~ address
  user = User.find_by_address(address)
  return true if user.nil?
  not user.use_anti_spam?
end

define_envelope_recipient_stopper do |context, address|
  stop_apply?(address[1..-2])
end
```

まとめ: 得意を活かす

- ✓ インターフェイス: Ruby
 - ✓ **直感的**設定方法
 - ✓ **柔軟な**動作変更
- ✓ コア: 高速なメール処理

Rubyインタプリタ埋込で
柔軟なアプリケーション

話したこと

2.times do |i|

✓ CとRubyの得手不得手

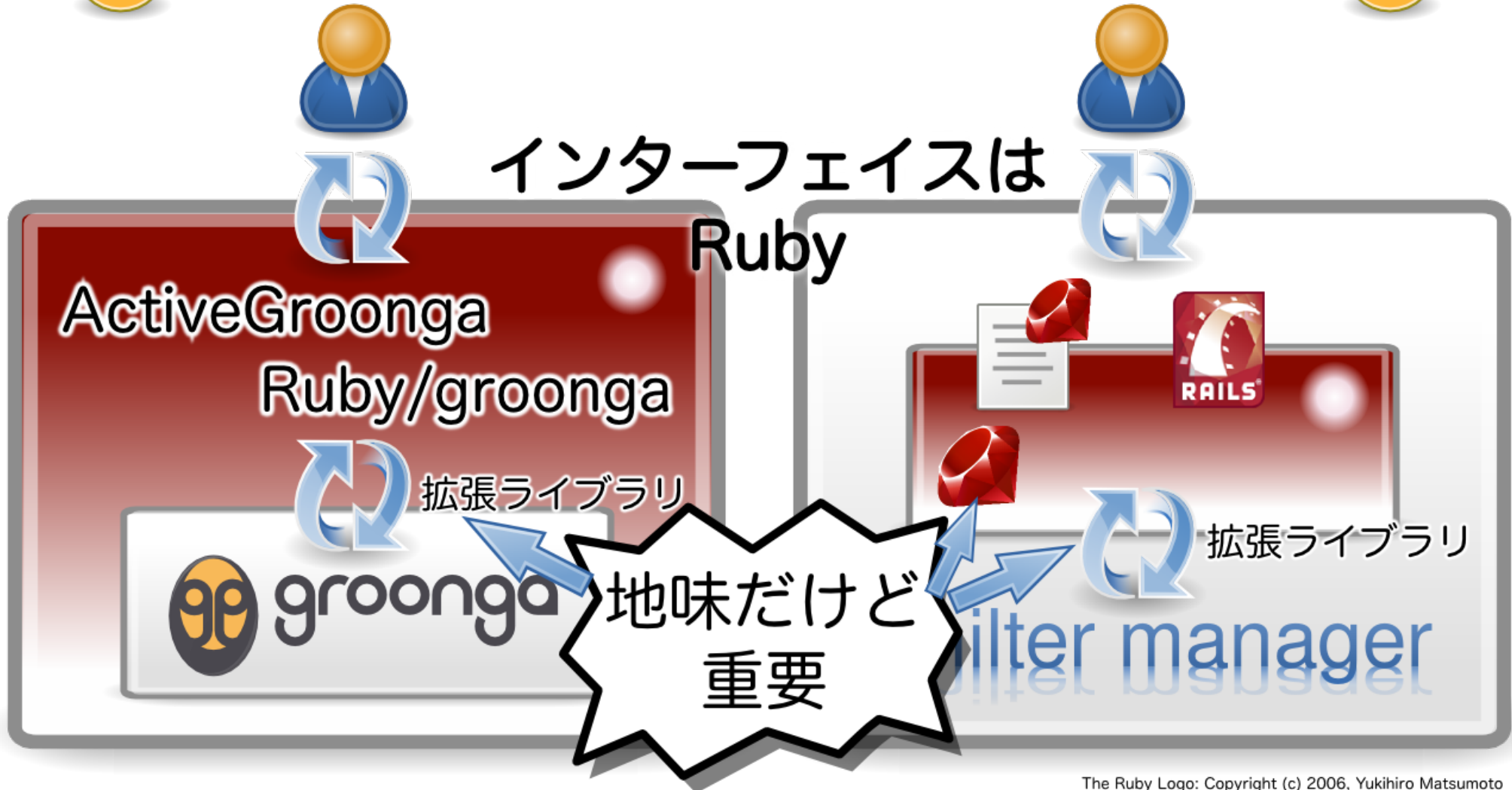
✓ よいとこどりをするには

end

まとめ

😊 苦手をカバー

得意を活かす 😊



The Ruby Logo: Copyright (c) 2006, Yukihiro Matsumoto

クリアコード

間のごことはクリアコードへ:



ClearCode



The Ruby Logo: Copyright (c) 2006, Yukihiro Matsumoto

✓ 詳しくはブースで