

Ruby on Rails

それは何か？

歴史

<http://wiki.fdiary.net/rails/?RailsHistory>

2004年7月 最初のリリース

2004年？

- 時期は遅い
 - J2EE (Java陣営)
 - WebObjects (NeXT→Apple)
 - ASP.NET
- なぜ？
 - PHPに対する不満らしい
- Rubyとの出会い
 - Rubyならきれいに書ける
 - Rubyには「本格的な=MVC」フレームワークが不在だった

DRY Principle

<http://c2.com/cgi/wiki?DontRepeatYourself>

*Every piece of **knowledge** must have a single, unambiguous, authoritative representation within a system.*

— [Andrew Hunt](#)

Rails

- Rails
 - ActiveSupport
 - ActionPack
 - ActiveRecord
 - ActionWebservice
 - ActionMailer
- WEBrick
- Rake
- RubyGems
- Ruby

Rails: 21th Century's Way of development

- J2EE Blueprint
 - MVC (JSP type2) Framework
- PofEAA
 - ActiveRecord
 - (more)
- Pragmatic Programmer
 - DRY
- Agile
 - テストファースト、漸進的設計.....



多層

エンタープライズコンピューティング

- クライアントーサーバ ×
- Web3層 (1996～1997年)

プレゼンテーション

ビジネスロジック

データ

→ Windows DNA、J2EE、(WebObjects)

スクリプト

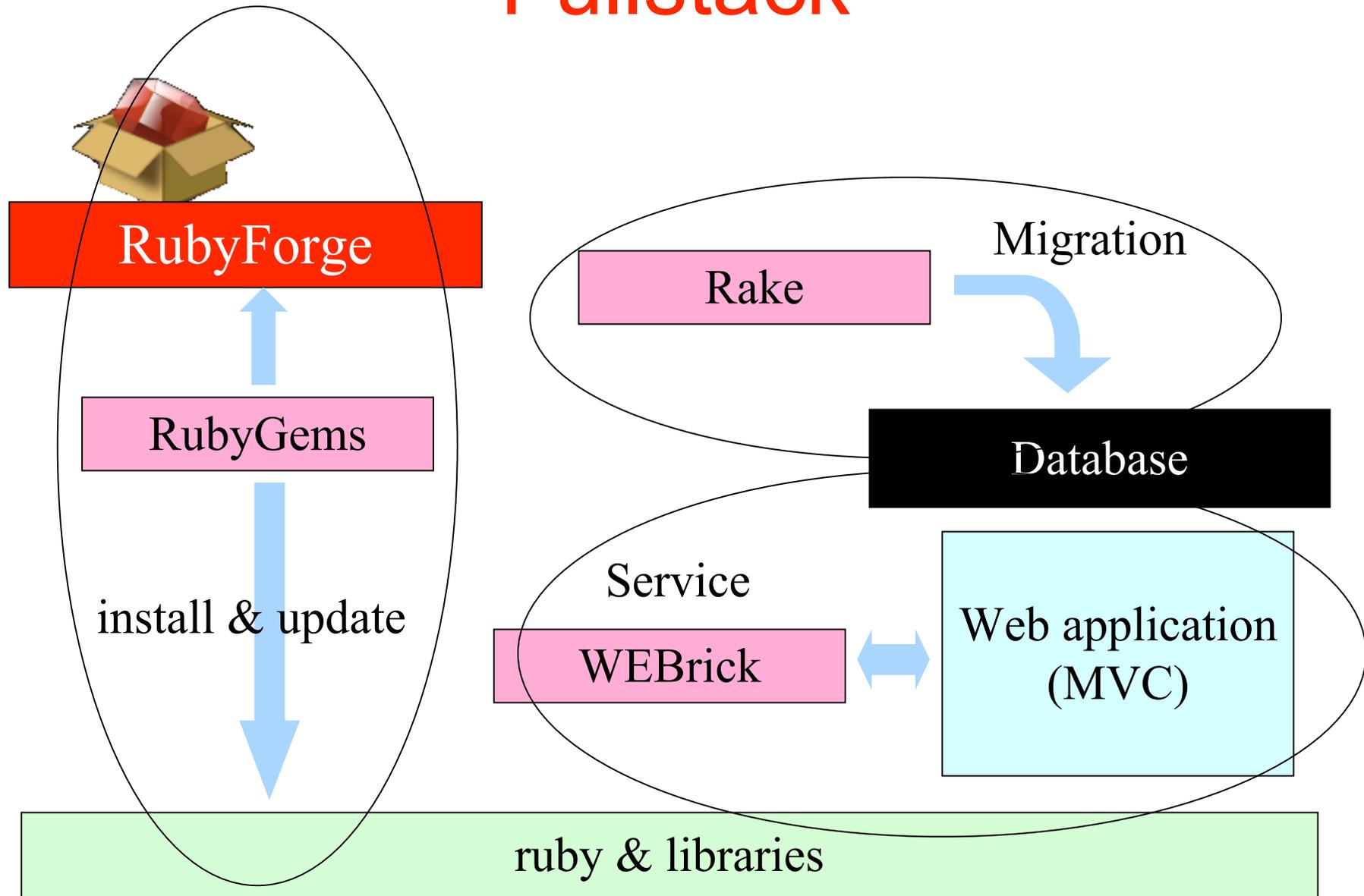
CGI (1993~1994 NCSA HTTPd)

- Perl
- PHP (+ DBアクセス) (1996~1997)

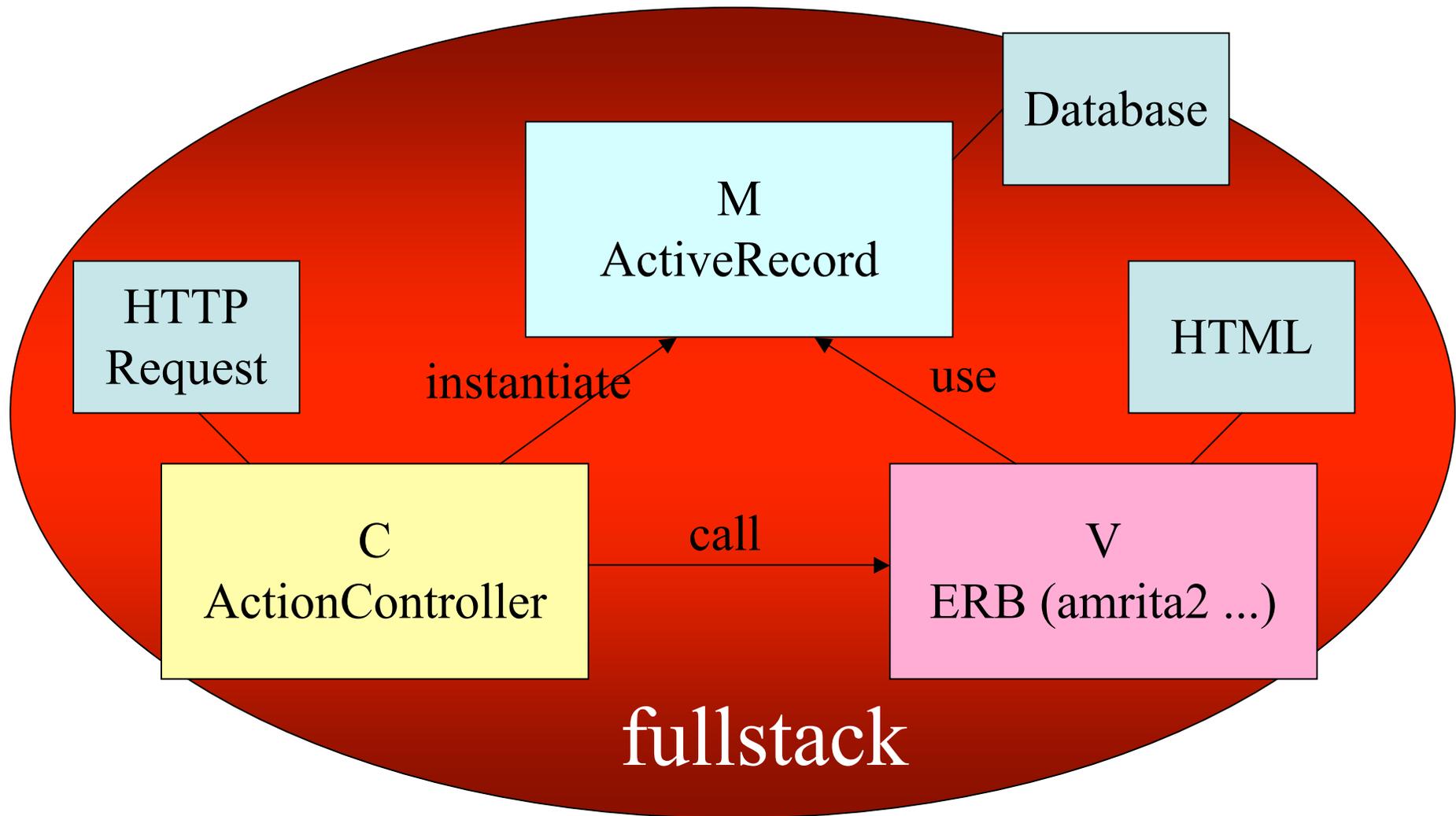
シンプル

→ 1990年代前半のVBを
1990年代後半に再現している.....

Fullstack



MVC (!= GUI := Web) フレームワーク



コードサンプル(M)

```
class Item < ActiveRecord::Base  
end
```

コードサンプル(C)

```
class ItemsController <
  ApplicationController
  def show
    @item = Item.find(params[:id])
  end
end
```

コードサンプル(V)

```
<table>
<% for column in Item.content_columns %>
<tr><td><%= column.human_name %></td>
  <% if column.name == 'url' %>
    <td> <a href="<%=h @item.url %>"><%=h @item.url %></a> </td>
  <% else %>
    <td> <%=h @item.send(column.name) %> </td>
  <% end %>
</tr>
<% end %>
</table>
<%= link_to 'Edit', :action => 'edit', :id => @item %> |
<%= link_to 'Back', :action => 'list' %>
```

Migration

- ソース → Subversion
- データベース → ?

漸進的設計の補助

コードサンプル(migrate)

```
class CreateTable < ActiveRecord::Migration
  def self.up
    create_table(:items) do |table|
      table.column :name , :string, :limit => 32
      table.column :url, :string, :limit => 128
      table.column :created_at, :timestamp
      table.column :memo, :text
    end
  end

  def self.down
    drop_table :items
  end
end
```

DRY

script/generate

コードの雛形生成

Controller

Model

View

Migration

DSL

Domain Specific Language (ドメイン特化言語)

<http://capsctrl.que.jp/kdmsnr/wiki/bliki/?DomainSpecificLanguage>

`table.column :name , :string, :limit => 32`

- メソッド呼び出しの()の省略

`table.column(:name, :string, :limit =>32)`

- 文字列の代替としてのシンボル型

`table.column("name", "string", "limit" => 32)`

- オプション引数としてのHash

`table.column("name", ... Hash.new["limit"] = 32)`

開発の流れ(例)

- プロジェクト生成
 - railsコマンド
- モデル生成
 - script/generate model
- テーブル作成
 - rake migrate
- コントローラ/ビュー生成
 - script/generate controller ...
- サーバー起動
 - script/server

plugin

<http://wiki.rubyonrails.org/rails/pages/Plugins>

```
script/plugin discover
```

```
script/plugin install
```

```
script/plugin list
```

南木氏の労作(pluginリスト)

<http://blog.netswitch.jp/articles/tag/plugin>

Ajax

- prototype.js
- effects.js
- dragdrop.js
- controls.js

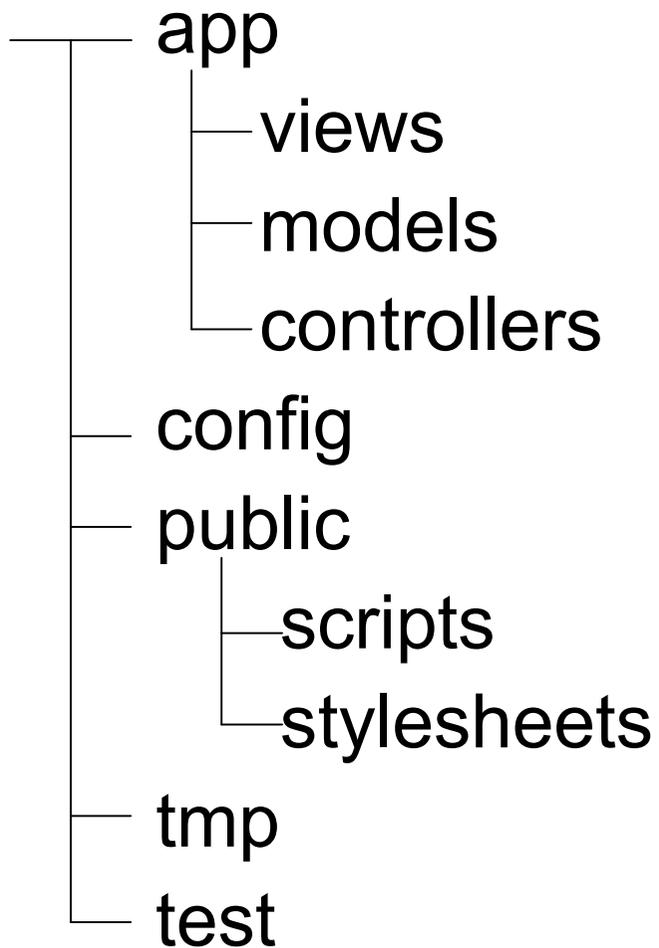
- **ActionView**
 - link_to_remote
 - submit_to_remote
 - observes

Convention over Configuration

規約があれば設定は不要（お好みの訳でどうぞ）

- 設定できるが、決めてある

CoC:ディレクトリ構成



CoC:テーブルとクラス

テーブル名 (複数形)

items

クラス名 (単数形)

Item

コレクション名 (複数形 : ActiveRecord has_many)

items

(ActiveSupport Inflector)

CoC:テーブル

カラム名

id: プライマリーキー

item_id: 外部キー(itemsテーブル)

出版ラッシュ

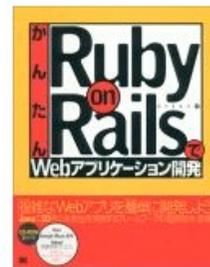
- Ruby界の幅の広さ(多様性は善)

实用(リファレンス)

- アジャイル開発
- オブジェクト指向
- ビジネス
- Web2.0

個人

事業



入門(チュートリアル)

バイブル

Railsによる

アジャイル

Webアプリケーション開発

オーム社

<http://www.loudthinking.com/>

<http://blogs.pragprog.com/cgi-bin/pragdave.cgi>

<http://shugo.net/jit/>



現場のRails

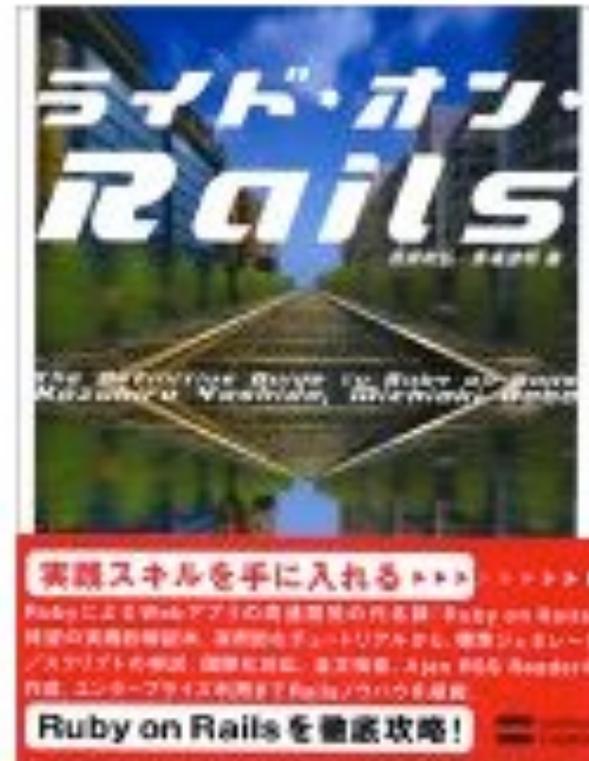
ライド・オン・Rails

Ruby on Railsを徹底攻略

ソフトバンク

<http://moriq.tdiary.net/>

<http://d.hatena.ne.jp/babie/>



アジャイル開発

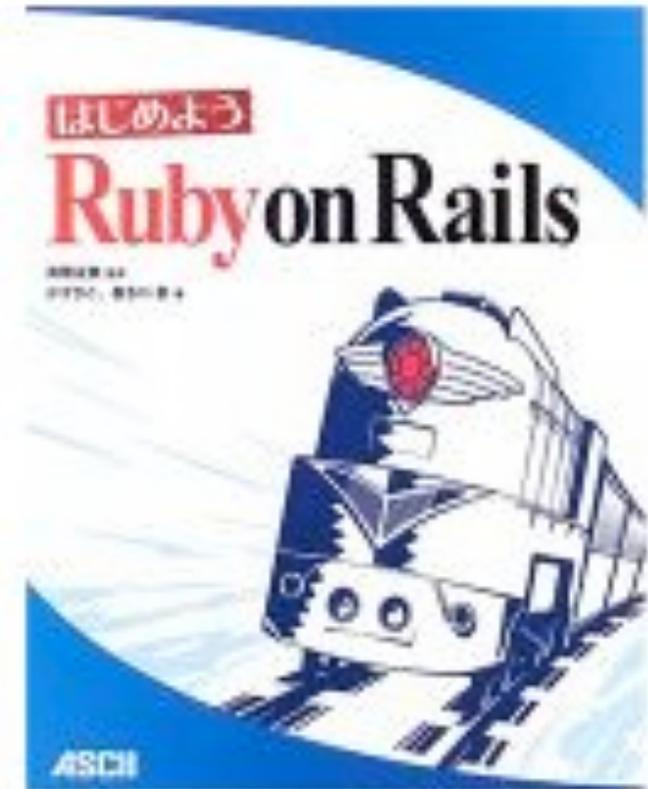
はじめよう

Ruby on Rails

アスキー

<http://kazuhiko.tdiary.net/>

<http://d.hatena.ne.jp/takahashim/>

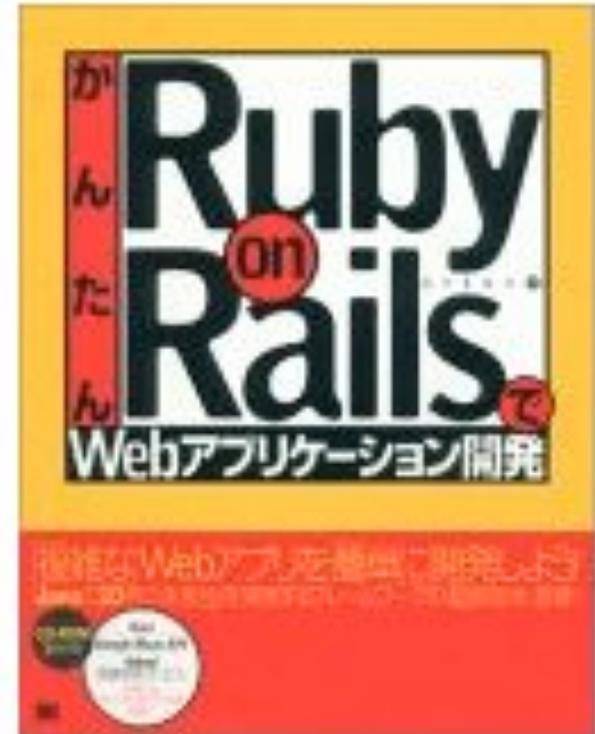


趣味プログラミング

かんたん Ruby on Railsで Webアプリケーション開発

翔泳社

<http://arton.no-ip.info/diary/>



オブジェクト指向&リファレンス

優しいRails

の育て方 (という題かどうかはわかりません)

＼(・∀・)ノくまくまー

<http://wota.jp/ac/>

レシピ集

?

<http://d.hatena.ne.jp/moro/>

<http://wiki.fdiary.net/rails/?RailsMeetingTokyo-000>

Rails for Designers

デザイナーのためのRails

(題名未定)

デザイナーがRailsを使う時に必要な情報を解説

のりおさん

<http://toykyo.net/>

URL

<http://www.rubyonrails.org>

日本語情報 & 日本語ML

<http://wiki.fdiary.net/rails/>