RUBY ON RAILS 2.1

WHAT'S NEW ?

CARLOS BRANDO REVIEW: MARCOS TAPAJÓS - COVER: DANIEL LOPES

Ruby on Rails 2.1

WHAT'S NEW

Second Edition

Ruby on Rails 2.1

WHAT'S NEW

Second Edition

Carlos Brando
Marcos Tapajós

Second edition: June 2008

Carlos Brando
Website: www.nomedojogo.com

Marcos Tapajós
Website: www.improveit.com.br/en/company/tapajos

Chapter 1

Introdução

Por volta do mês de julho de 2004 David Heinemeier Hansson lançou publicamente o framework Ruby on Rails, que havia sido extraído de um software chamado Basecamp. Mais de três anos depois, no dia 7 de dezembro de 2007 o Ruby on Rails chegou a sua versão 2.0 com diversas alterações importantes.

De lá para cá se passaram seis meses, e neste tempo mais de **1400 programadores** do mundo todo contribuiram criando **1600 patches**. E hoje, 1 de junho de 2008 o Ruby on Rails chega à sua versão 2.1.

De acordo com David as principais novidades nesta versão são:

- Timezones
- Dirty tracking
- Gem Dependencies
- Named scope
- UTC-based migrations

- Better caching

Para atualizar ou instalar a nova versão, é o de sempre:

```
gem install rails
```

AGRADECIMENTOS

Ao Marcos Tapajós que é o co-autor deste livro. Se não fosse por ele acho que você não estaria lendo isto.

Ao Daniel Lopes que fez uma linda capa para esta edição.

A toda a comunidade brasileira de Ruby on Rails que colaborou direta ou indiretamente com este livro, comentando os textos no blog e dando sugestões. É como sempre costumo dizer, o melhor do Rails é a comunidade! Continuem criando, inventando e principalmente compartilhando...

Chapter 2

ActiveRecord

O Active Record é uma camada de mapeamento objeto-relacional (object-relational mapping layer), responsável pela interoperabilidade entre a aplicação e o banco de dados e pela abstração dos dados. (wikipedia)

Chapter 3

ActiveSupport

Active Support é uma coleção de várias classes úteis e extensões de bibliotecas padrões, que foram considerados úteis para aplicações em Ruby on Rails. (wikipedia)

INTRODUZINDO MEMOIZABLE PARA CACHE DE ATRIBUTOS

Performance é coisa séria, e um dos métodos mais usados para aumentar a velocidade de execução em códigos é o uso de cache. Quem nunca fez algo assim?

```
class Person < ActiveRecord::Base
  def age
    @age ||= um_calculo_muito_complexo
  end
end
```

Nesta versão do Rails temos uma forma mais elegante de fazer isto usando o método **memoize** (é **memoize** mesmo e não **memorize**). Vamos alterar o exemplo acima para funcionar com esta nova funcionalidade:

```
class Person < ActiveRecord::Base
  def age
    um_calculo_muito_complexo
  end
  memoize :age
end
```

O método **age** será executado apenas uma vez e o seu retorno será armazenado e retornado em futuras chamadas ao método.

Só existe uma diferença entre os dois códigos acima. No primeiro, como o método é executado todas as vezes, se o valor armazenado na variável **@age** for **nil** ou **false** o cálculo (muito complexo) será executado novamente até termos a idade da pessoa.

No segundo exemplo, o método **age** só será executado uma vez e o valor retornado será sempre devolvido nas próximas chamadas, mesmo que seja **nil** ou **false**.

Chapter 4

ActiveResource

O ActiveResource é uma camada de mapeamento responsável pela implementação do lado cliente de sistemas RESTful. Através do ActiveResource é possível consumir serviços RESTful através do uso de objetos que funcionam como um proxy para serviços remotos.

Chapter 5

ActionPack

Compreende o Action View (geração de visualização de usuário, como HTML, XML, JavaScript, entre outros) e o Action Controller (controle de fluxo de negócio). (wikipedia)

CACHES_ACTION

Foi acrescentado a opção **:layout** no método **caches_action**.

```
class ListsController < ApplicationController
  ...

  caches_action :index, :layout => false

  ...
end
```

No exemplo acima eu especifiquei **:layout => false**, isto significa que o layout não será armazenado no cache, apenas o conteúdo da action será. Isto é muito útil quando temos conteúdo dinâmico no layout (o que acontece na maioria dos casos).

Se você não especificar nada ele assumirá o padrão atual que é **true**.

RJS#PAGE.RELOAD

O método **reload** foi incluído ao **ActionView::Helpers::PrototypeHelper** para ser usado em templates **.rjs** ou blocos **render(:update)**. Este método força a recarga da página atual no browser usando javascript. Em outras palavras é um atalho para o já muito usado **window.location.reload();**.

Veja como usar:

```
respond_to do |format|
  format.js do
    render(:update) { |page| page.reload }
  end
end
```

Chapter 6

ActionController

O ActionController é a camada responsável por receber as requisições web e de tomar as decisões do quê será executado e renderizado ou de redirecionar para outra ação. Uma ação é definido como métodos públicos nos controladores que são automaticamente disponíveis através das rotas.

Chapter 7

ActionView

O ActionView é a camada responsável pela geração da interface visível ao usuário através da conversão dos templates ERB.

Chapter 8

Railties

Chapter 9

Rake Tasks, Plugins e Scripts

Chapter 10

Prototype e script.aculo.us

Chapter 11

Ruby 1.9

Chapter 12

Debug

Chapter 13

Bugs e Correções

END_OF_QUARTER

Nem bem havia saído o Rails 2.1 e já foi encontrado um erro sério. Se você ainda tiver um projeto criado nesta versão entre no **irb** e tente rodar isto:

```
Date.new(2008, 5, 31).end_of_quarter
```

ERRO!

Por que? A implementação do método **end_of_quarter** foi feita da maneira errada, ele avança até o último mês do trimestre e depois pega último dia. O problema é que ele apenas avança o mês, e como estou partindo do dia 31 de maio, ele tentar criar uma nova instância do objeto **Date** para 31 de junho, que não existe. Com o objeto **Time** não é disparado uma exceção, mas ele retorna a data errada: 31 de julho.

Nesta versão este erro já foi corrigido, mas caso você ainda esteja usando a versão 2.1 em algum projeto, muito cuidado, porque este erro só ocorrerá se usarmos o método **end_of_quarter** nos dias 31 de maio, julho e agosto.

Chapter 14

Informações Adicionais

Chapter 15

CHANGELOG

RUBY ON RAILS 2.1

WHAT'S NEW?

CARLOS BRANDO REVIEW: MARCOS TAPAJÓS - COVER: DANIEL LOPES