Rails 2.0

Michael Koziarski

Introduction

2.0 is pretty old

Fri Dec 7 13:37:13 2007 +0000

8330

2.1 Beta 1

2008-04-01

What's In 2.0?

Less Cruft

Unused Functionality

Dead End Technology

Dead End Technology

SOAP

Orphans

Evolutionary Enhancements

Enhancements in 2.0

Resource Routes

Resource Routes

```
map.resources :posts
```

Resource Routes

```
map.resources :posts
```

http://rails.example.com/posts/1

Resource Routes

```
map.resources :posts do |posts|
  posts.resources :comments
  posts.resources :trackbacks
end
```

Resource Routes

```
map.resources :posts do |posts|
  posts.resources :comments
  posts.resources :trackbacks
end
```

http://rails.example.com/posts/1/comments

Resource Routes

```
map.resources :posts do |posts|
  posts.resources :comments
  posts.resources :trackbacks
end
```

http://rails.example.com/posts/1/comments

http://rails.example.com/posts/1/comments.xml

Resource Routes

```
map.resources :posts,
    :has_many => [:comments, :trackbacks]
```

Resource Routes

```
map.namespace :admin do |admin|
  admin.resources :posts, :has_many => [:comments]
end
```

Resource Routes

```
map.namespace :admin do |admin|
  admin.resources :posts, :has_many => [:comments]
end
```

http://rails.example.com/admin/posts/1/comments.json

Record Identification

simply_helpful

Resources

```
map.resources :people
```

Resources

http://rails.example.com/people
http://rails.example.com/people/1

Resources

```
class PeopleController < ApplicationController
  def show
    @person = Person.find(params[:id])
  end

  def index
    @people = Person.find(:all)
  end
end
```

Resources

```
link_to @person.name,
        person_url(@person)
```

What About Forms?

Resources

```
form_for :person, @person,
  :url  => person_url(@person),
  :html => {:method=>:put} do |f|
```

Resources

```
form_for :person, @person,
         :url => {:action => :update}
```

Resources

```
form_for :person, Person.new,
  :url  => people_url,
  :html => {:method=>:post} do |f|
```

Resources

```
form_for(@person) do |f|
```

Resources

```
link_to @person.name, @person
```

Resources

```
redirect_to @person
```

Resources

```
form_for([@post, @comment])
```

```
link_to [:admin, @post, @comment]
```

Plugin Extractions

Pagination

```
@person_pages, @people =
  paginate :people, :order => 'last_name'
```

Will Paginate

```
@people = @account.people.paginate
```

acts_as_.*

acts_as_.*

- acts_as_nested_set

acts_as_.*

- acts_as_nested_set
- acts_as_tree

acts_as_.*

- acts_as_nested_set
- acts_as_tree
- acts_as_list

acts_as_list

```ruby
class Comment < ActiveRecord::Base
  acts_as_list :scope=>:post
end
```

```ruby
@comment.move_to_bottom
@other_comment = @comment.higher_item
@comment.in_list?
```

Foxy Fixtures

Fixtures

```
koz:
    id: 34
    name:  Michael Koziarski
    email: michael@koziarski.com
```

Fixtures

```
hello_world:
    id: 45
    author_id: 34
    body: '<p>Hello World </p>'
    created_at: <%= 1.month.ago.to_s(:db) %>
    updated_at: <%= 1.month.ago.to_s(:db) %>
```

Fixtures

```
troll_on_hello_world:
    id: 4
    post_id: 45
    body: FRIST POST!!!!11
```

Fixtures

```
hello_world:
    id: 45
    author_id: 34
    body: '<p>Hello World </p>'
    created_at: <%= 1.month.ago.to_s(:db) %>
    updated_at: <%= 1.month.ago.to_s(:db) %>
```

Fixtures

```
hello_world:
    id: 45
    author_id: 34
    body: '<p>Hello World </p>'
    created_at: <%= 1.month.ago.to_s(:db) %>
    updated_at: <%= 1.month.ago.to_s(:db) %>
```

Fixtures

```
hello_world:
    id: 45
    author_id: 34
    body: '<p>Hello World </p>'
    created_at: <%= 1.month.ago.to_s(:db) %>
    updated_at: <%= 1.month.ago.to_s(:db) %>
```

Fixtures

```
hello_world:
    id: 45
    author_id: 34
    body: '<p>Hello World </p>'
    created_at: <%= 1.month.ago.to_s(:db) %>
    updated_at: <%= 1.month.ago.to_s(:db) %>
```

Fixtures

```
hello_world:
    id: 45
    author_id: 34
    body: '<p>Hello World </p>'
    created_at: <%= 1.month.ago.to_s(:db) %>
    updated_at: <%= 1.month.ago.to_s(:db) %>
```

Fixtures

```yaml
hello_world:
    id: 45
    author_id: 34
    body: '<p>Hello World </p>'
    created_at: <%= 1.month.ago.to_s(:db) %>
    updated_at: <%= 1.month.ago.to_s(:db) %>
```

Fixtures

```
hello_world:
    id: 45
    author_id: 34
    body: '<p>Hello World </p>'
    created_at: <%= 1.month.ago.to_s(:db) %>
    updated_at: <%= 1.month.ago.to_s(:db) %>
```

Fixtures

```
hello_world:
    author: koz
    body: '<p>Much Simpler</p>'
```

Fixtures

```yaml
hello_world:
    author: koz
    body: '<p>Tagged</p>'
    tags: cool, rails
```

Security Enhancements

XSS

XSS

```
<% @comments.each do |comment| %>
  <p><%= comment.body %></p>
<% end %>
```

XSS

```
<script>
    alert("haxxed!!");
</script>
```

XSS

<plaintext>

XSS

```
<% @comments.each do |comment| %>
  <p><%=h comment.body %></p>
<% end %>
```

XSS

```
<% @comments.each do |comment| %>
  <p><%= sanitize(comment.body) %></p>
<% end %>
```

Sanitize

Sanitize

- Old version was black-list based

Sanitize

- Old version was black-list based
- Allowed several obscure but dangerous attacks

Sanitize

- Old version was black-list based
- Allowed several obscure but dangerous attacks
- white_list plugin to the rescue!

XSS

```
<% @comments.each do |comment| %>
  <p><%= sanitize(comment.body) %></p>
<% end %>
```

XSS

```
<% @comments.each do |comment| %>
    <p>
        <%= sanitize(comment.body,
                tags => %w(b i em a)) %>
    </p>
<% end %>
```

CSRF

AKA Session Riding

How to defeat digg.com

... an introduction to session riding

http://4diggers.blogspot.com

```
<img src="http://yoursite.com/sites/add_vote/54" />
```

CSRF Prerequisites

- ‘Remember Me’
- Guessable URLs

POST won’t save you

```
<form id="hax" action="http://yoursite.com"
                method="POST">
</form>
```

```
<form id="hax" action="http://yoursite.com"
                method="POST">
</form>

<script>$('hax').submit()</script>
```

CSRF

```
class PostsController < ApplicationController
  protect_from_forgery
end
```

Performance

Date Parsing

Active Record

Attributes

Named Routes

Named Routes

```
map.foo 'foo', :controller=>"foos"
```

Named Routes

```
foo_url # => "http://rails.example.com/foo"
```

Named Routes

```
person_url(1)
```

Named Routes

```
person_url(1, :show_full=>true)
```

HTTP Authentication

```ruby
authenticate_or_request_with_http_basic do |username, password|
  User.authenticate(username, password)
end
```

Sexy Migrations

```ruby
create_table :customers do |t|
  t.column :first_name, :string
  t.column :last_name,  :string
  t.column :account_id, :integer
  t.column :created_at, :datetime
  t.column :updated_at, :datetime
end
```

```ruby
create_table :customers do |t|
  t.string :first_name, :last_name
  t.references :account
  t.timestamps
end
```

Initializers

Initializers

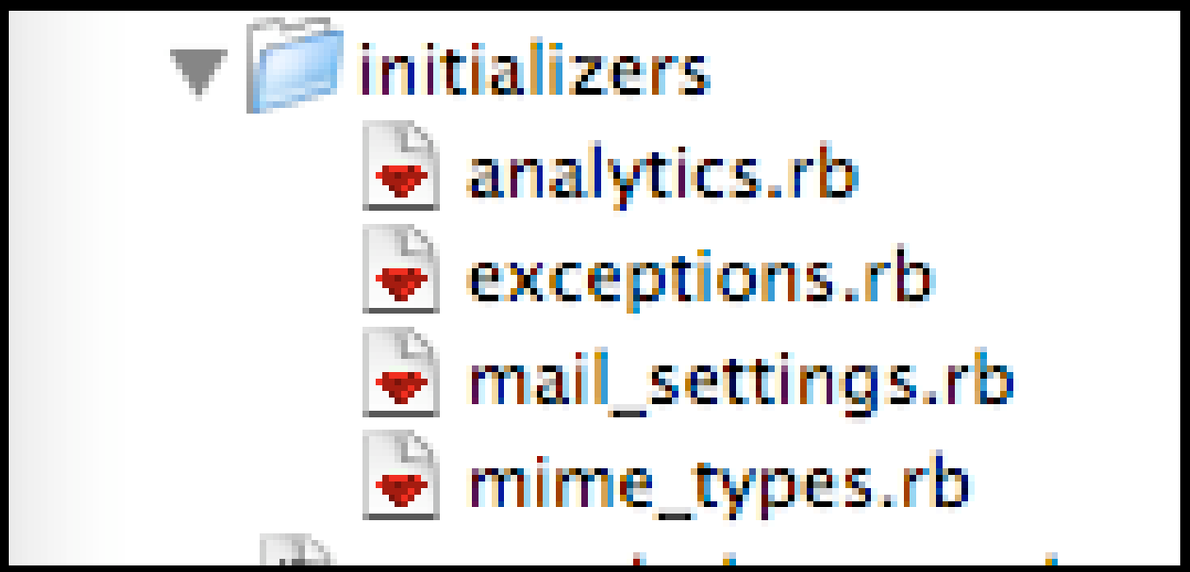
```
Inflector.inflections do |inflect|
  inflect.irregular 'koz', 'kozes'
end

require 'some_gem'
require 'some_other_gem'

SOME_CONSTANT = SomeGem.new(:something)
```

Initializers

Initializers



Coming in 2.1

named_scope

Named Scopes

```
class Article < ActiveRecord::Base
  named_scope :published,
      :conditions => {:published => true}
  named_scope :popular,
      :conditions =>  "views_count > 50"
end
```

Named Scopes

```
Article.published.paginate(:page => 1)
```

Named Scopes

```
Article.published.popular.count
```

Named Scopes

```
@blog.articles.published.
      popular.paginate(:page=>1)
```

Association Preloading

Like :include, but not

Preloading

```
Post.find(:all, :include=>:comments)
```

Preloading

```
Post.find(:all, :include=>[:comments,
                           :authors])
```

Preloading

```
Post.find(:all, :include=>[:comments,
                           {:authors=>:preferences}])
```

More Performance

Route Recognition

Memory Optimisations

Conclusions

Incremental

Enhancement

Targeted Optimisations

Less Mass

Questions?

Michael Koziarski <michael@koziarski.com>