

**MVC**

**ORM**

**ROR**

Yann Klis, 2016

# Ruby ON Rails

**Model**

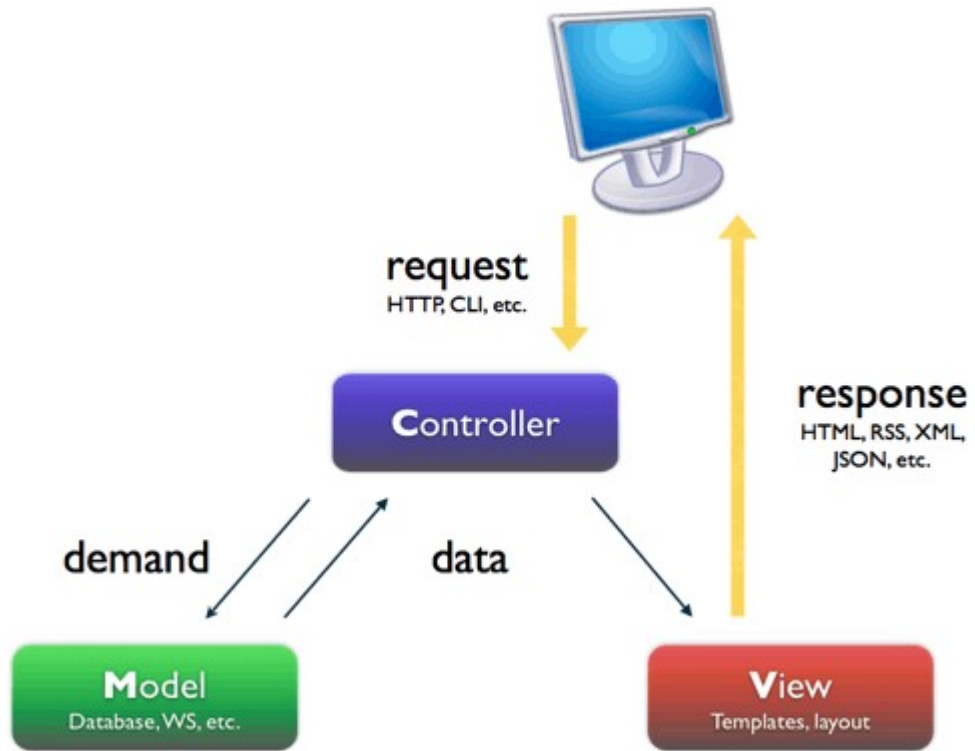
**View**

**Controller**

**M**odel

**V**iew

**C**ontroller



**Convention  
over  
Configuration**

Convention  
**over**  
Configuration

first\_app

FAVORITES

DEVICES

Name	Date Modified	Size	Kind
app	Today 2:39 PM	--	Folder
assets	Today 2:32 PM	--	Folder
images	Today 2:32 PM	--	Folder
javascripts	Today 2:32 PM	--	Folder
stylesheets	Today 2:32 PM	--	Folder
controllers	Today 2:32 PM	--	Folder
application_controller.rb	Today 2:32 PM	80 bytes	Ruby Source
helpers	Today 2:32 PM	--	Folder
mailers	Today 2:32 PM	--	Folder
models	Today 2:32 PM	--	Folder
views	Today 2:32 PM	--	Folder
layouts	Today 2:32 PM	--	Folder
config	Today 2:39 PM	--	Folder
config.ru	Today 2:32 PM	158 bytes	Document
db	Today 2:32 PM	--	Folder
doc	Today 2:32 PM	--	Folder
Gemfile	Today 2:32 PM	708 bytes	Document
Gemfile.lock	Today 2:33 PM	3 KB	Document
lib	Today 2:32 PM	--	Folder
log	Today 2:32 PM	--	Folder
public	Today 2:32 PM	--	Folder
Rakefile	Today 2:32 PM	273 bytes	Document
README	Today 2:32 PM	9 KB	Document
script	Today 2:32 PM	--	Folder
test	Today 2:32 PM	--	Folder
tmp	Today 2:32 PM	--	Folder
vendor	Today 2:32 PM	--	Folder



# environments

```
# ls -l config/environments/  
development.rb  
production.rb  
test.rb
```

**DRY**

**DRY**

Don't Repeat Yourself

# ActiveRecord

ORM

M in MVC

app/models/product.rb

```
class Product < ActiveRecord::Base
end
```

db/migrate/20130626120509\_create\_products.rb

```
class CreateProducts < ActiveRecord::Migration
  def change
    create_table :products do |t|
      t.string :name
      t.text :description

      t.timestamps
    end
  end
end
```

# ActiveRecord / migrations

```
mysql> show tables;
```

```
+-----+  
| Tables_in_monapppli_development |  
+-----+  
| products                         |  
| schema_migrations                |  
+-----+
```

```
2 rows in set (0.00 sec)
```

```
mysql> describe schema_migrations;
```

```
+-----+-----+-----+-----+-----+-----+  
| Field      | Type                | Null | Key | Default | Extra |  
+-----+-----+-----+-----+-----+-----+  
| version    | varchar(255)        | NO   | PRI | NULL     |       |  
+-----+-----+-----+-----+-----+-----+
```

```
1 row in set (0.00 sec)
```

```
mysql> select * from schema_migrations;
```

```
+-----+  
| version          |  
+-----+  
| 20130626120509  |  
+-----+
```

```
1 row in set (0.00 sec)
```

```
irb(main):001:0> product = Product.new
```

```
=> #<Product id: nil, name: nil, description: nil, created_at: nil, updated_at: nil>
```

```
irb(main):002:0> product.name = "Macbook"
```

```
=> "Macbook"
```

```
irb(main):003:0> product.save
```

```
(0.4ms) BEGIN
```

```
SQL (0.3ms) INSERT INTO `products` (`created_at`, `name`, `updated_at`) VALUES  
(`2013-06-26 12:08:17`, 'Macbook', '2013-06-26 12:08:17')
```

```
(4.7ms) COMMIT
```

```
=> true
```

```
irb(main):004:0> Product.count
```

```
(0.5ms) SELECT COUNT(*) FROM `products`  
=> 1
```

```
irb(main):005:0> Product.first
```

```
Product Load (0.5ms) SELECT `products`.* FROM `products` ORDER BY  
`products`.`id` ASC LIMIT 1  
=> #<Product id: 1, name: "Macbook", description: nil, created_at: "2013-06-26  
12:08:17", updated_at: "2013-06-26 12:08:17">
```

```
irb(main):006:0> Product.where(name: "Macbook")
```

```
Product Load (0.8ms) SELECT `products`.* FROM `products` WHERE  
`products`.`name` = 'Macbook'  
=> #<ActiveRecord::Relation [#<Product id: 1, name: "Macbook", description: nil,  
created_at: "2013-06-26 12:08:17", updated_at: "2013-06-26 12:08:17">]>
```

```
irb(main):008:0> Product.where(name: "Macbook").first
```

```
Product Load (0.7ms) SELECT `products`.* FROM `products` WHERE  
`products`.`name` = 'Macbook' ORDER BY `products`.`id` ASC LIMIT 1  
=> #<Product id: 1, name: "Macbook", description: nil, created_at: "2013-06-26  
12:08:17", updated_at: "2013-06-26 12:08:17">
```



> Product.where(name: "Macbook")

```
Product Load (0.8ms) SELECT `products`.* FROM `products` WHERE  
`products`.`name` = 'Macbook'
```

```
=> #<ActiveRecord::Relation [#<Product id: 1, name: "Macbook", description: nil,  
created_at: "2013-06-26 12:08:17", updated_at: "2013-06-26 12:08:17">]>
```

> Product.where(name: "Macbook").to\_a

```
Product Load (0.4ms) SELECT `products`.* FROM `products` WHERE  
`products`.`name` = 'Macbook'
```

```
=> [#<Product id: 1, name: "Macbook", description: nil, created_at: "2013-06-26  
12:08:17", updated_at: "2013-06-26 12:08:17">]
```

app/models/product.rb

```
class Product < ActiveRecord::Base
  validates_presence_of :name
end
```

db/migrate/20130626120509\_create\_products.rb

```
class CreateProducts < ActiveRecord::Migration
  def change
    create_table :products do |t|
      t.string :name
      t.text :description

      t.timestamps
    end
  end
end
```

```
> product = Product.new
```

```
=> #<Product id: nil, name: nil, description: nil, created_at: nil, updated_at: nil>
```

```
> product.save
```

```
(0.3ms) BEGIN
```

```
(0.2ms) ROLLBACK
```

```
=> false
```

```
> product.errors
```

```
=> #<ActiveModel::Errors:0x007f2f9fb1bec8 @base=#<Product id: nil, name: nil, description: nil, created_at: nil, updated_at: nil>, @messages={:name=>["can't be blank"]}>
```

app/models/product.rb

```
class Product
  validates_presence_of :name
  belongs_to :category
end
```

app/models/category.rb

```
class Category
  has_many :products
end
```

app/models/product.rb

```
class Product

  before_validate :reformat

  before_save :check_name

  after_save :rebuild_category

  after_destroy :destroy_category_if_empty

end
```

# ActiveModel

ActiveModel::Model

ActiveModel::Callbacks

ActiveModel::Validations

ActiveModel::Dirty

ActiveModel::AttributeMethods

ActiveModel::Serialization

ActiveModel::Translation

# Routing

**Restful**



<b>Verbe HTTP</b>	<b>Url</b>	<b>Action</b>
<b>GET</b>	/articles	Liste des articles
<b>POST</b>	/articles	Création d'un article
<b>GET</b>	/articles/new	Formulaire de création d'un article
<b>GET</b>	/articles/:id/edit	Formulaire d'édition d'un article
<b>GET</b>	/articles/:id	Visualisation d'un article
<b>PUT</b>	/articles/:id	Mise à jour d'un article
<b>DELETE</b>	/articles/:id	Suppression d'un article

**asset**

**pipeline**

# asset pipeline

app/assets/javascripts/application.js

```
//= require jquery
//= require jquery_ujs
//= require jquery-ui
//= require bootstrap
//= require bootstrap-datepicker
//= require jquery_nested_form
//= require ckeditor/init
//= require_tree .
```

# asset pipeline

app/assets/stylesheets/application.css.scss

```
*= require "custombootstrap"  
*= require "fontawesome"  
*= require "datepicker"  
*= require_self
```

**ActiveSupport**

```
> "Je suis à la gare".blank ?  
=> false
```

```
> "".present?  
=> false
```

```
> "Je suis à la gare".parameterize  
=> "je-suis-a-la-gare"
```



```
> "John était allongé à la table d'un  
café quand soudain le loup  
apparu.".truncate(22)  
=> "John était allongé ..."
```

**ActionView**

# ActionView

```
products.html.erb  
products.html.haml  
movies.json.erb  
movies.xml.builder
```

# Test

Tests unitaires des models, des controllers, des helpers  
Tests fonctionnels (rspec, capybara, etc.)

**rails console**

> **Product.where(name: "Macbook")**

Product Load (0.8ms) SELECT `products`.\* FROM `products` WHERE  
`products`.`name` = 'Macbook'

=> #<ActiveRecord::Relation [#<Product id: 1, name: "Macbook", description: nil,  
created\_at: "2013-06-26 12:08:17", updated\_at: "2013-06-26 12:08:17">]>

> **Product.where(name: "Macbook").to\_a**

Product Load (0.4ms) SELECT `products`.\* FROM `products` WHERE  
`products`.`name` = 'Macbook'

=> [#<Product id: 1, name: "Macbook", description: nil, created\_at: "2013-06-26  
12:08:17", updated\_at: "2013-06-26 12:08:17">]

**rails server**

**rails generate**



tâches rake

debugger

i18n

# Communauté Ruby

**Une certaine mentalité, certes...**

**rubygems.org**

repository centralisé des gems



**a gem**  
**There's ~~an App~~**  
**For That**

# devise

authentication

# omniauth

omniauth-facebook, omniauth-twitter,  
omniauth-dailymotion, omniauth-github, ...



# paperclip

upload de fichiers, stockage, thumbnails, etc.

**will\_paginate**

pagination

# state\_machine

machine à états...

# airbrake

gestion des exceptions

# resque

gestion des jobs asynchrones  
(backend Redis)

# Apprendre Rails

<http://railsforzombies.org/>

<http://guides.rubyonrails.org/>