



DOLPH LUNDGREN

THE CONTROLLER

BIG BROTHER IS NO LONGER JUST WATCHING....HE'S KILLING

NU IMAGE PRESENTS A TWO STICKS / ROUGH DIAMOND / METROPOLITAN MEDIA PRODUCTION A FILM BY SIDNEY FURIE
DOLPH LUNDGREN 'THE CONTROLLER' HANNEA JAEINCKE ROBERT MAND PAUL CAMPSON CO-PRODUCED BY R.C. FILON BILL KRAVITZ
EXECUTIVE PRODUCERS AN LERNER DANNY DIMICORT TREVOR SHORT BOAZ DAVIDSON
PRODUCED BY RICHARD SALVATORE JULIA VERNON DAVID CRONSTON JOHN MAPPIN STORY BY JOHN MAPPIN JULIA VERNON
WRITTEN BY R.C. FILON JULIA VERNON JOHN MAPPIN DIRECTED BY SIDNEY FURIE

NO. 1 CHOICE

ME

A horizontal bar composed of four colored segments: yellow, purple, yellow, and red.

Guille Carlos

@guillec

Bitpop

Ruby on Rails

Android

Burritos

A vertical purple bar on the left side of the slide.

THE CONTROLLER

MVC

- The C in MVC
- The middle wo/man
- Make sense of request and produce correct output.

The Controller in Rails

- Its a class
- It has methods
- It inherits from ApplicationController

The Controller in Rails

- Application receives request
- Routing determines which controller and action to call
- Rails creates a instance of that controller and runs the method that matches action

EXAMPLE: WAKE UP PEOPLE

request: /authors/new

- ac = AuthorsController.new
- ac.new

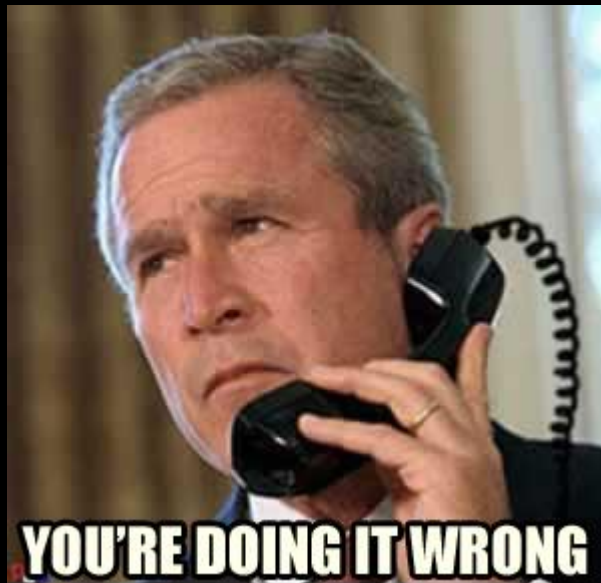
```
class AuthorsControllers < ApplicationController
  def new
  end
end
```

You have seen this:

```
class ClientsController < ApplicationController
  def new
  end

  def create
  end

  def update
  end
end
```



WHATCHOO TALKIN 'BOUT



WILLIS

Single Responsibility Principle (SRP)

Not to be confused with SPR



SEXY PUERTO RICAN

SRP

A decorative graphic consisting of a horizontal bar at the top and a vertical bar on the left, both made of four colored segments: yellow, purple, yellow, and red. The yellow segment on the left is thicker and has a rounded corner at the top.

A class should have one and only one reason to change.

WAT

- If a class assumes more than one responsibility, then there will be more than one reason for it to change.
- If a class has more than one responsibility, then the responsibilities become coupled.
- Changes to one responsibility may impair or inhibit the class' ability to meet the others.
- This kind of coupling leads to fragile designs that break in unexpected ways when changed.

Focused Controllers

Cool project by @jonleighton on GitHub

Concept:

Change Rails conventions. Rather than controllers being classes that contain one method per action, controllers are now namespaces and each action is a class within that namespace.

```

module PostsController
  # Action is a common superclass for all the actions
  # inside `PostsController`.
  class Action < ApplicationController
    include FocusedController::Mixin
  end

  class Index < Action
    def run
      # Your code here.
    end

    # No instance variables are shared with the view. Instead,
    # public methods are defined.
    def posts
      @posts ||= Post.all
    end

    # To prevent yourself having to write `controller.posts`
    # in the view, you can declare the method as a helper
    # method which means that calling `posts` automatically
    # delegates to the controller.
    helper_method :posts
  end

  # Actions do not need to declare a `run` method - the default
  # implementation inherited from `FocusedController::Mixin` is an
  # empty method.
  class Show < Action
    def post
      @post ||= Post.find params[:id]
    end
    helper_method :post
  end
end

```

References

- SRP: <http://www.objectmentor.com/resources/articles/srp.pdf>
- Focused Controllers:
https://github.com/jonleighton/focused_controller
- SPR:
[@guillec](#)