# Rails 3 Ropes Course


Gregg Pollack

Nathaniel Bibler

Thomas Meeks

Jacob Swanner

Tyler Hunt

Mark Kendall

Caike Souza


envylabs

# Rails 3 Ropes Course

**Getting Started & Routes**

Workshop - Lab #1

**Bundler & ActionController**

Workshop - Lab #2

**ActionMailer**

Workshop - Lab #3

**ActiveRelation & ActiveModel**

Workshop - Lab #4

**XSS & UJS**

Workshop - Lab #5

envylabs

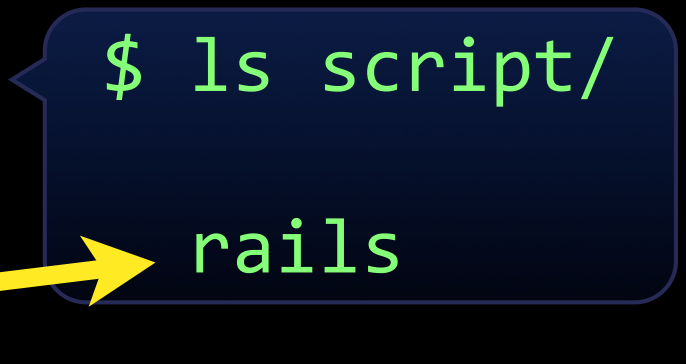
# Starting a New App

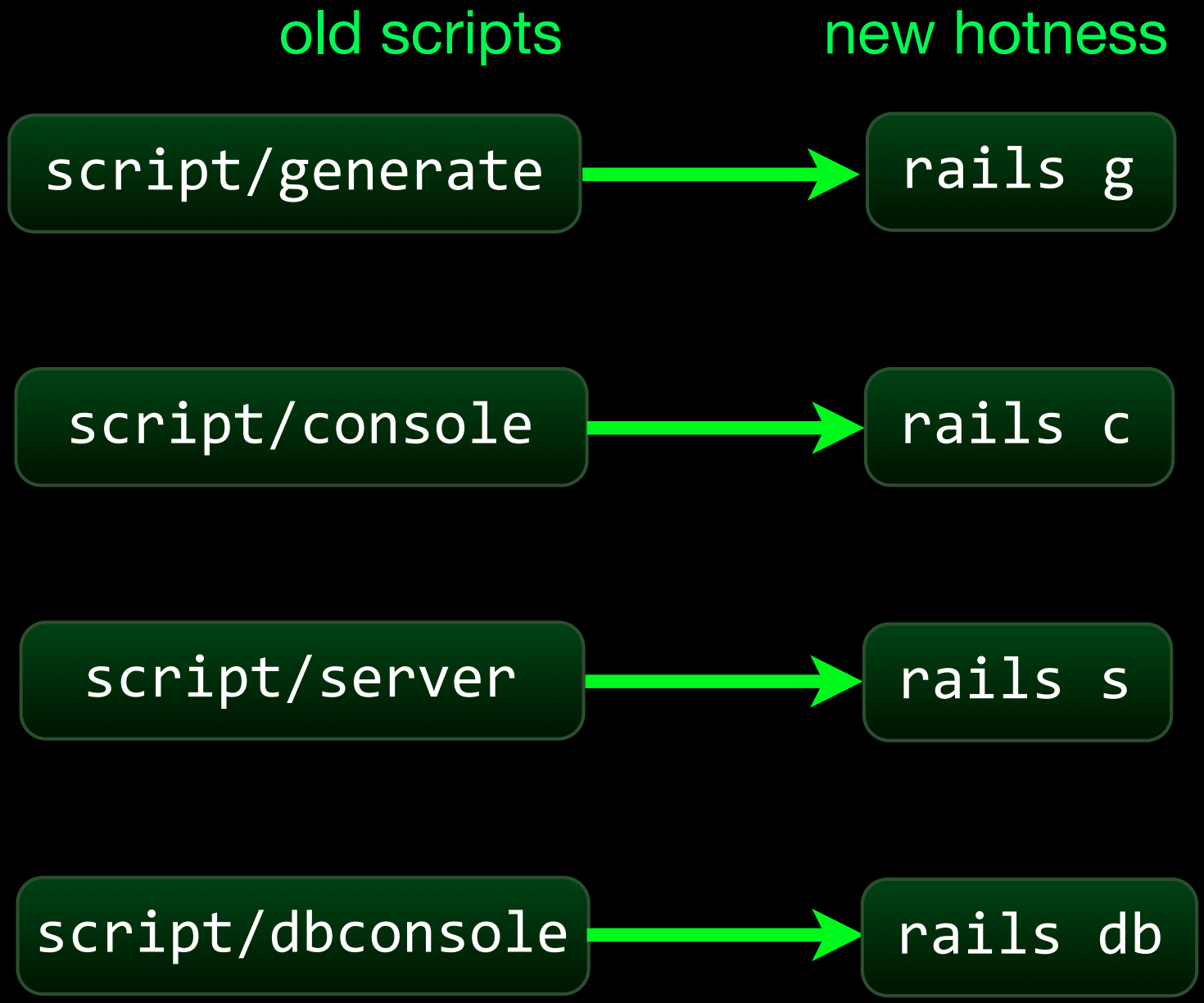
```
$ rails
Usage:
  rails APP_PATH [options]

Options:
  -r, [--ruby=PATH]           # Path to the Ruby binary of your choice
  -d, [--database=DATABASE]   # Preconfigure for selected database
  -m, [--template=TEMPLATE]   # Path to an application template
      [--dev]                 # Setup the application with
                              # Gemfile pointing to your Rails checkout
      [--edge]                # Setup the application with
                              # Gemfile pointing to Rails repository
      [--skip-gemfile]        # Don't create a Gemfile
  -O, [--skip-activerecord]   # Skip ActiveRecord files
  -T, [--skip-testunit]       # Skip TestUnit files
  -J, [--skip-prototype]      # Skip Prototype files
  -G, [--skip-git]            # Skip Git ignores and keeps
```

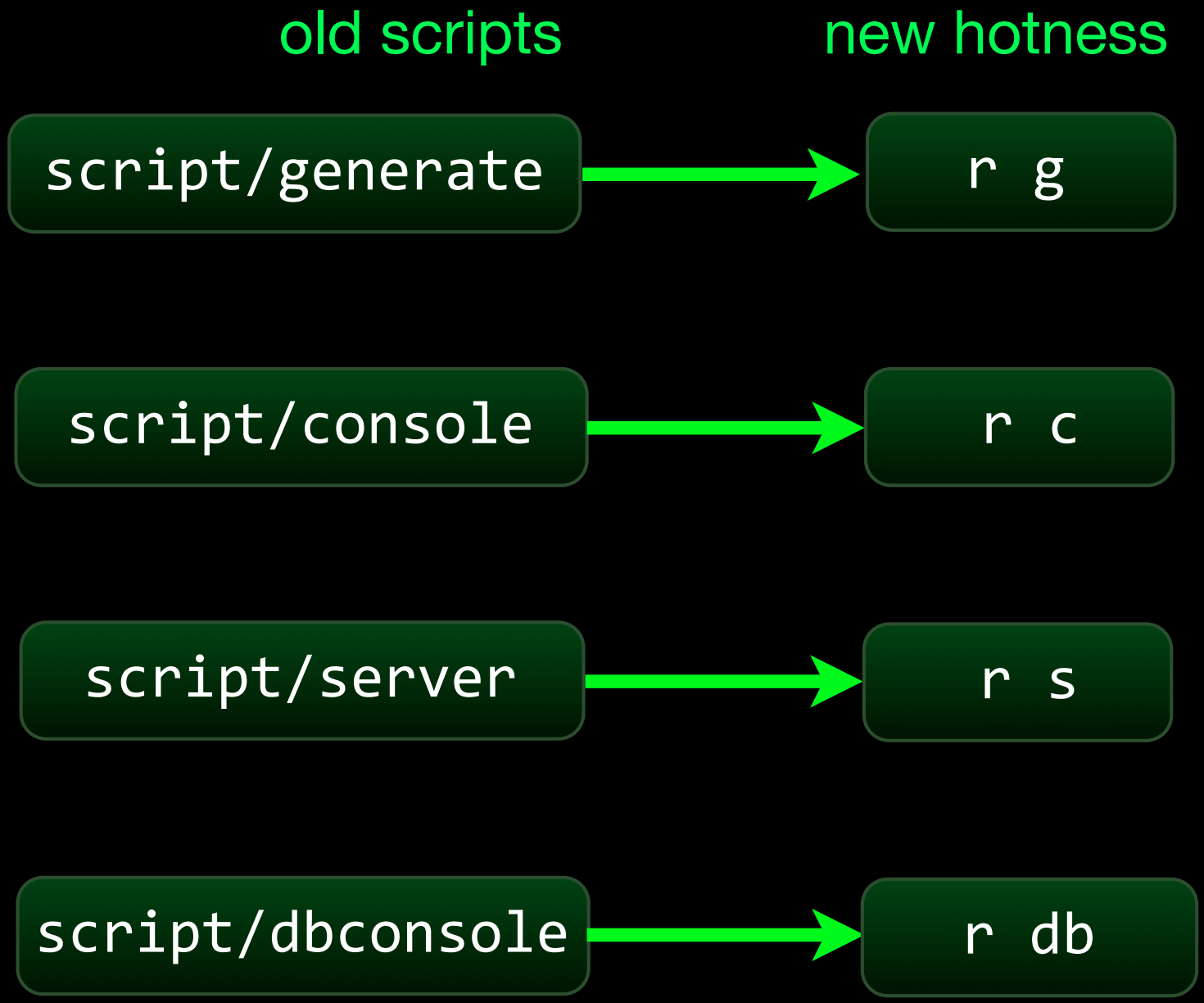
.gitignore

```
.bundle
db/*.sqlite3
log/*.log
tmp/**/*
```

```
$ rails test_app
      create
      create  README
      create  .gitignore
      ...
$ cd test_app/
$ rails
Usage: rails COMMAND [ARGS]

The most common rails commands are:
 generate    Generate new code (short-cut alias: "g")
 console     Start the Rails console (short-cut alias: "C")
 server      Start the Rails server (short-cut alias: "S")
 dbconsole   Start a console for the database specified in config/database.yml
             (short-cut alias: "db")

In addition to those, there are:
 application  Generate the Rails application code
 destroy      Undo code generated with "generate"
 benchmarker  See how fast a piece of code runs
 profiler     Get profile information from a piece of code
 plugin       Install a plugin
 runner       Run a piece of code in the application environment

All commands can be run with -h for more information.
```




old scripts
new hotness
script/generate
rails g
script/console
rails c
script/server
rails s
script/dbconsole
rails db


old scripts
new hotness
script/generate
r g
script/console
r c
script/server
r s
script/dbconsole
r db
alias r='rails'

config/environment.rb

Rails 2

```
Rails::Initializer.run do |config|

  config.load_paths += %W( #{RAILS_ROOT}/extras )

  config.gem "bj"
  config.gem "sqlite3-ruby", :lib => "sqlite3"
  config.gem "aws-s3", :lib => "aws/s3"

  config.plugins = [ :exception_notification ]

  config.time_zone = 'UTC'

end
```

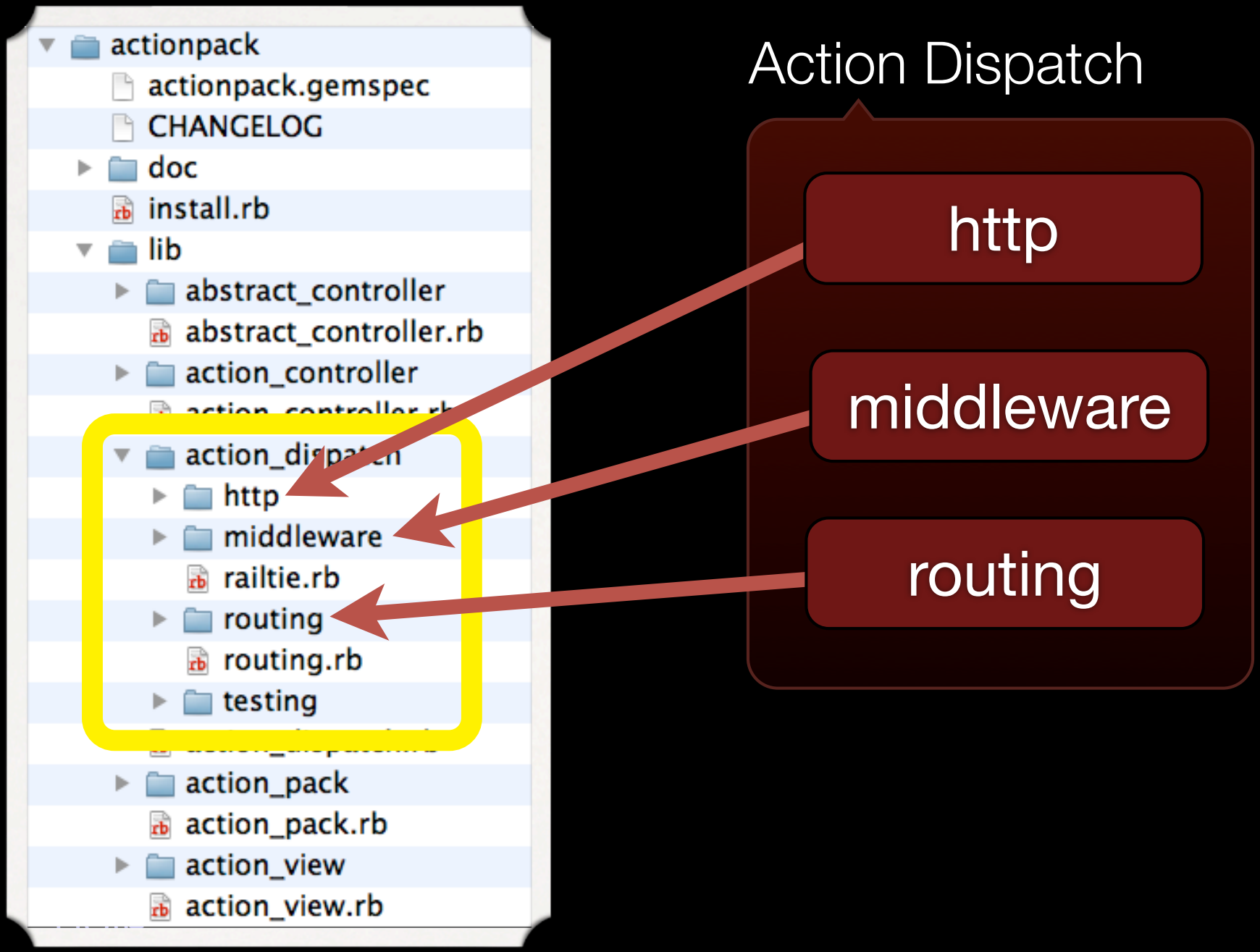
config/application.rb

Rails 3

```
module TestApp
  class Application < Rails::Application

    config.load_paths += %W( #{RAILS_ROOT}/extras )

    config.plugins = [ :exception_notification ]

    config.time_zone = 'UTC'

  end
end
```

**It's a Rack Application**

# New Router API


actionpack
actionpack.gemspec
CHANGELOG
doc
install.rb
lib
abstract_controller
abstract_controller.rb
action_controller
action_dispatch
http
middleware
railtie.rb
routing
routing.rb
testing
action_pack
action_pack.rb
action_view
action_view.rb
Action Dispatch
http
middleware
routing

# New Routing API

config/routes.rb

**Rails 3**

```
TestApp::Application.routes.draw do |map|

end
```

# New Routing API

config/routes.rb

Rails 3

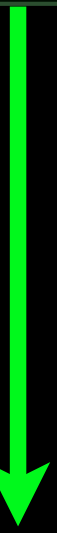
```
TestApp::Application.routes.draw do |map|

  map.resources :posts

end
```

**Old routing syntax works**

# New Routing API

**Rails 2**

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



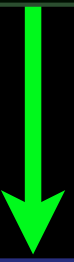
**Rails 3**

```
resources :posts do
  resources :comments
end
```

# New Routing API

**Rails 2**

```
post.resources :comments,
               :member => { :preview => :post },
               :collection => { :archived => :get }
```



**Rails 3**

```
resources :comments do
  member do
    post :preview
  end

  collection do
    get :archived
  end
end
```

**Rails 3**

```
resources :comments do
  post :preview, :on => :member
  get :archived, :on => :collection
end
```

Rails 2

```
map.connect 'login', :controller => 'session', :action => 'new'
```

Rails 3

```
match 'login' => 'session#new'
```

## Named Route `login_path`

Rails 2

```
map.login 'login', :controller => 'session', :action => 'new'
```

Rails 3

```
match 'login' => 'session#new', :as => :login
```

# Root Paths

Rails 2

```
map.root :controller => 'users', :action => 'index'
```

Rails 3

```
root :to => 'users#index'
```

## Legacy Route

Rails 2

```
map.connect ':controller/:action/:id'
map.connect ':controller/:action/:id.:format'
```

Rails 3

```
match ':controller(/:action(/:id(.:format)))'
```

(commented out by default)

# Optional Parameters

**Rails 2**

```
map.connect '/articles/:year/:month/:day', :controller => 'posts', :action => 'index'
```

**Rails 3**

```
match '/articles/:year/:month/:day' => "posts#index"
```

**Rails 2**

```
map.connect '/articles/:year/:month/:day', :controller => 'posts', :action => 'index'
map.connect '/articles/:year/:month', :controller => 'posts', :action => 'index'
map.connect '/articles/:year', :controller => 'posts', :action => 'index'
```

**Rails 3**

```
match '/articles(/:year(/:month(/:day)))' => "posts#index"
```

# Specifying the method

**Rails 2**

```
map.connect '/articles/:year', :controller => 'posts', :action => 'index',
                               :conditions => {:method => :get}
```

**Rails 3**

```
match '/articles/:year' => "posts#index", :via => :get
```

**Rails 3**

```
get '/articles/:year' => "posts#index"
```

# Redirection

**Rails 3**

```
match 'signin', :to => redirect("/login")
```

```
match 'users/:name', :to => redirect {|params| "/#{params[:name]}" }
```

```
match 'google' => redirect('http://www.google.com/')
```

# Constraints

**Rails 2**

```
map.connect '/:year', :controller => 'posts', :action => 'index',
                                  :requirements => { :year => /\d{4}/ }
```

**Rails 3**

```
match '/:year' => "posts#index", :constraints => {:year => /\d{4}/}
```

```
:constraints => { :user_agent => /iphone/ }

:constraints => { :ip => /192\.168\.1\.\d{1,3}/ }

constraints(:host => /localhost/) do
  resources :posts
end

constraints IpRestrictor do
  get 'admin/accounts' => "queenbee#accounts"
end
```

# Rack Goodness

Rails 3

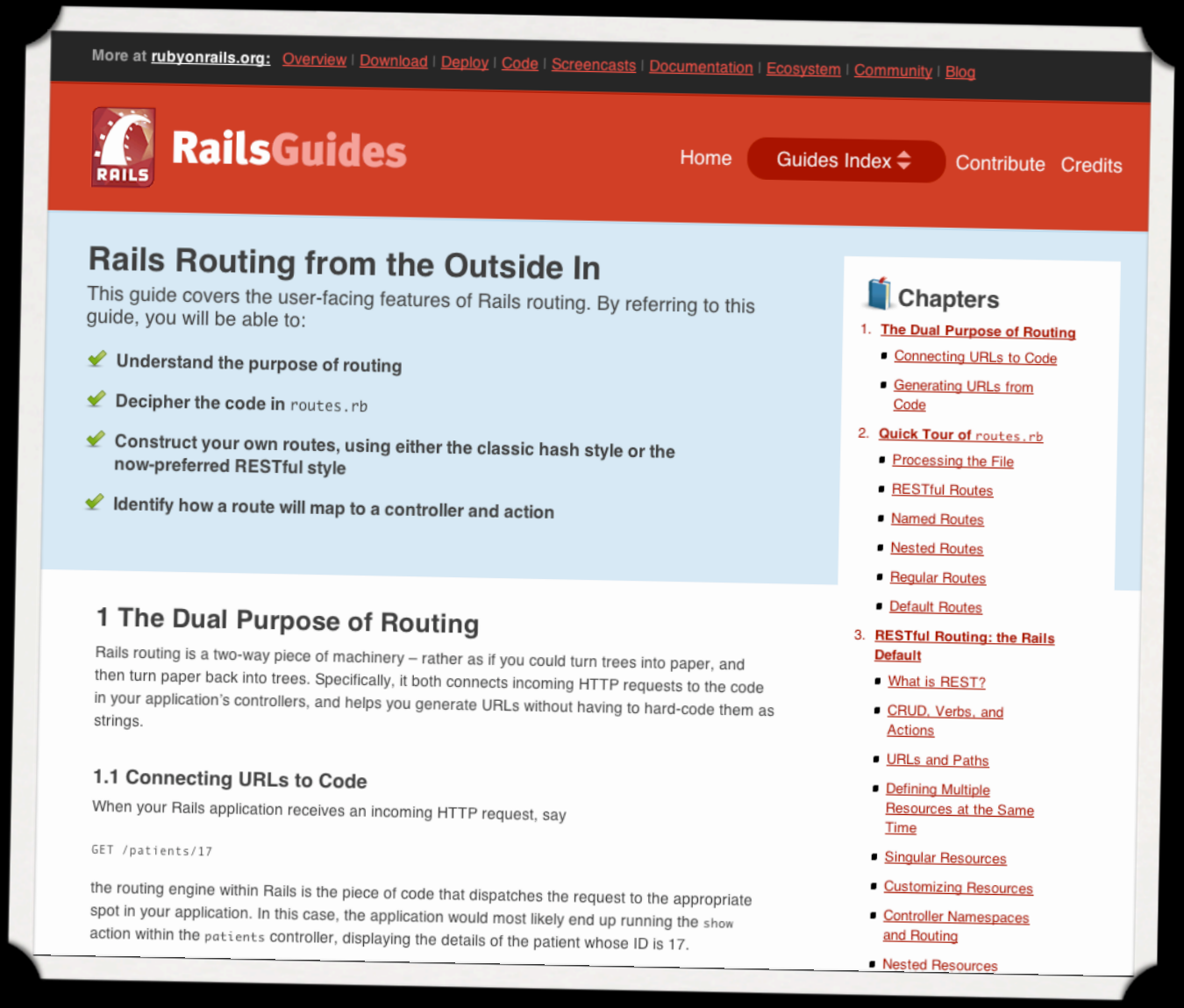
```
get 'hello' => proc { |env| [200, {},  "Hello Rack"] }
```

```
get 'rack_endpoint' => PostsController.action(:index)
```

```
get 'rack_app' => CustomRackApp
```

# For more information

## http://guides.rails.info/routing.html



# Tutorial - Lab #1

## Getting Started & Routes

Follow the directions in the README

# Rails 3 Ropes Course

- Getting Started & Routes
- Workshop - Lab #1
- **Bundler & ActionController**
- Workshop - Lab #2
- **ActionMailer**
- Workshop - Lab #3
- **ActiveRelation & ActiveModel**
- Workshop - Lab #4
- **XSS & UJS**
- Workshop - Lab #5

envylabs

# Dependency Management

# Typical Rails deployment

## Specify your gems inside environment.rb

```
config.gem "haml"
config.gem "chronic", :version => '0.2.3'
```

## $ rake gems:install

fetch, download, and install/compile these
gems into your system RubyGems directory

## $ rake gems:unpack:dependencies

unpacks the gems into your application
into your vendor/gems directory

# Issues

It's bound into Rails

Not great dependency resolution

Conflicts occur at runtime

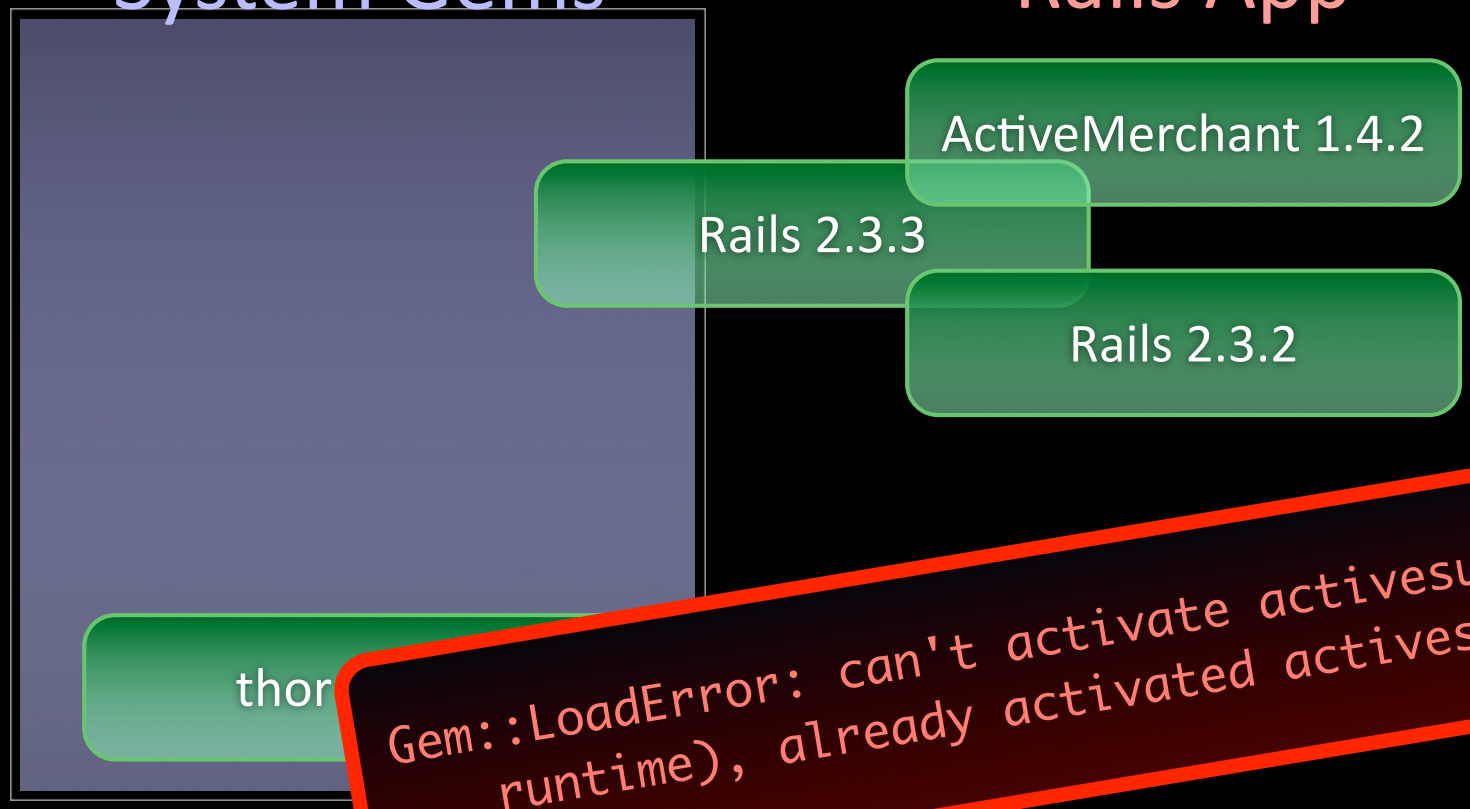
# Dependency Resolution

ActiveMerchant 1.4.2

activesupport >= 2.3.2

Rails 2.3.2

activesupport = 2.3.2



# Bundler Commands

Specify your gems inside Gemfile

```
gem "haml"
gem "chronic", '0.2.3'
```

$ bundle

fetch, download, and install/compile these gems

$ bundle package

Moves gem source into /vendor/cache

# With Bundler

ActiveMerchant 1.4.2

activesupport >= 2.3.2

Rails 2.3.2

activesupport = 2.3.2

## System Gems

Rails 2.3.3

thor 0.13.1

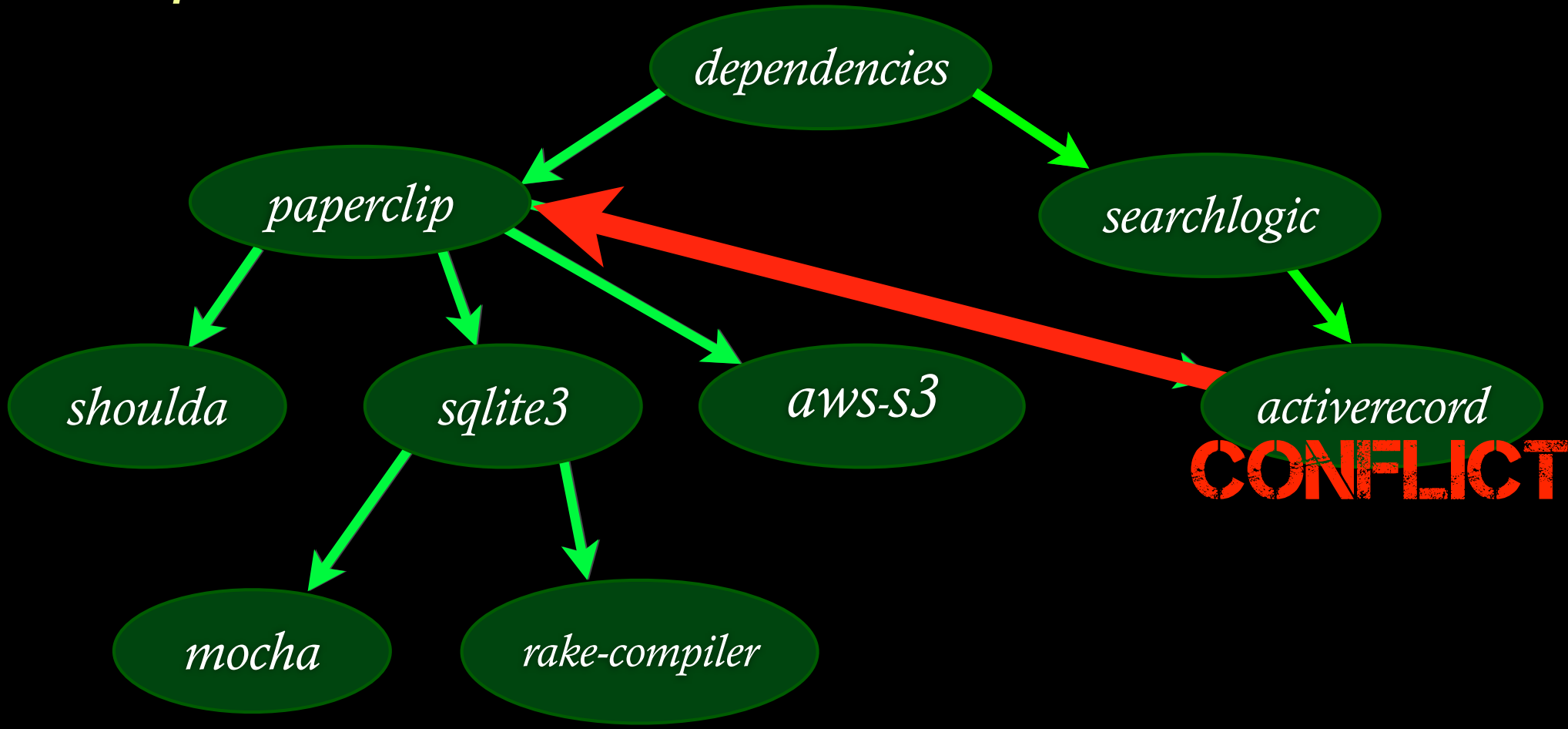
## Rails App

Rails 2.3.2

ActiveMerchant 1.4.2

# Dependency Resolution

## *Depth First Search*



# Gemfile Syntax

```
source "http://rubygems.org"

gem "hpricot", "0.6"
gem "sqlite3-ruby", :require => "sqlite3"
gem "rails", :git => "git://github.com/rails/rails.git"
gem "rails", :path => "~/Sites/rails"

git "git://github.com/rails/rails.git" do
  gem "railties"
  gem "active_model"
end

group :test do
  gem "webrat"
end
```

`$bundle`

Will ensure all gems are installed, including webrat (but it's only included in test mode).

`$bundle --without test`

Will install everything except webrat.

$ bundle check

Check to see if all the dependencies are available for the Gemfile

$ bundle show

Shows all libraries which are included by the Gemfile and their dependencies

$ bundle show <gem name>

Shows where the gem is located in our filesystem

$ bundle open <gem name>

Will open the gem source in our default editor

When I download a new application

I locally run

```
$ bundle
```

When I deploy my application

I run on the server

```
$ bundle
```

When my application is going live

Gemfile.lock → Check into Source Control

Specifies exact gem versions my application is running

on the server

```
$ bundle
```

Installs gems listed in Gemfile.lock rather then Gemfile

# Gemfile vs Gemfile.lock

## Gemfile

```
source 'http://rubygems.org'

gem 'capistrano'
gem 'nokogiri', '>=1.4.0'
gem 'sqlite3-ruby', :require => 'sqlite3'
```

## Gemfile.lock

```
GEM
  remote: http://rubygems.org/
  specs:
    capistrano (2.5.18)
      highline
      net-scp (>= 1.0.0)
      net-sftp (>= 2.0.0)
      net-ssh (>= 2.0.14)
      net-ssh-gateway (>= 1.0.0)
    nokogiri (1.4.0)
    sqlite3-ruby (1.2.5)
```

$ bundle

nokogiri 1.4.1 released!

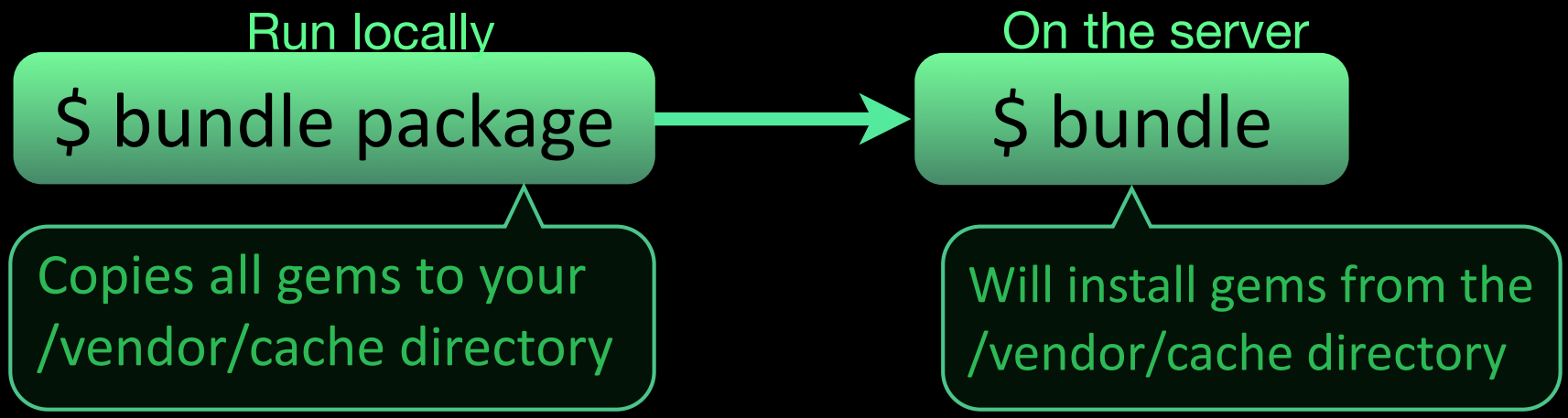
$ bundle

$ bundle update

Recreates the Gemfile.lock and runs “bundle” to install new dependencies

$ bundle update nokogiri

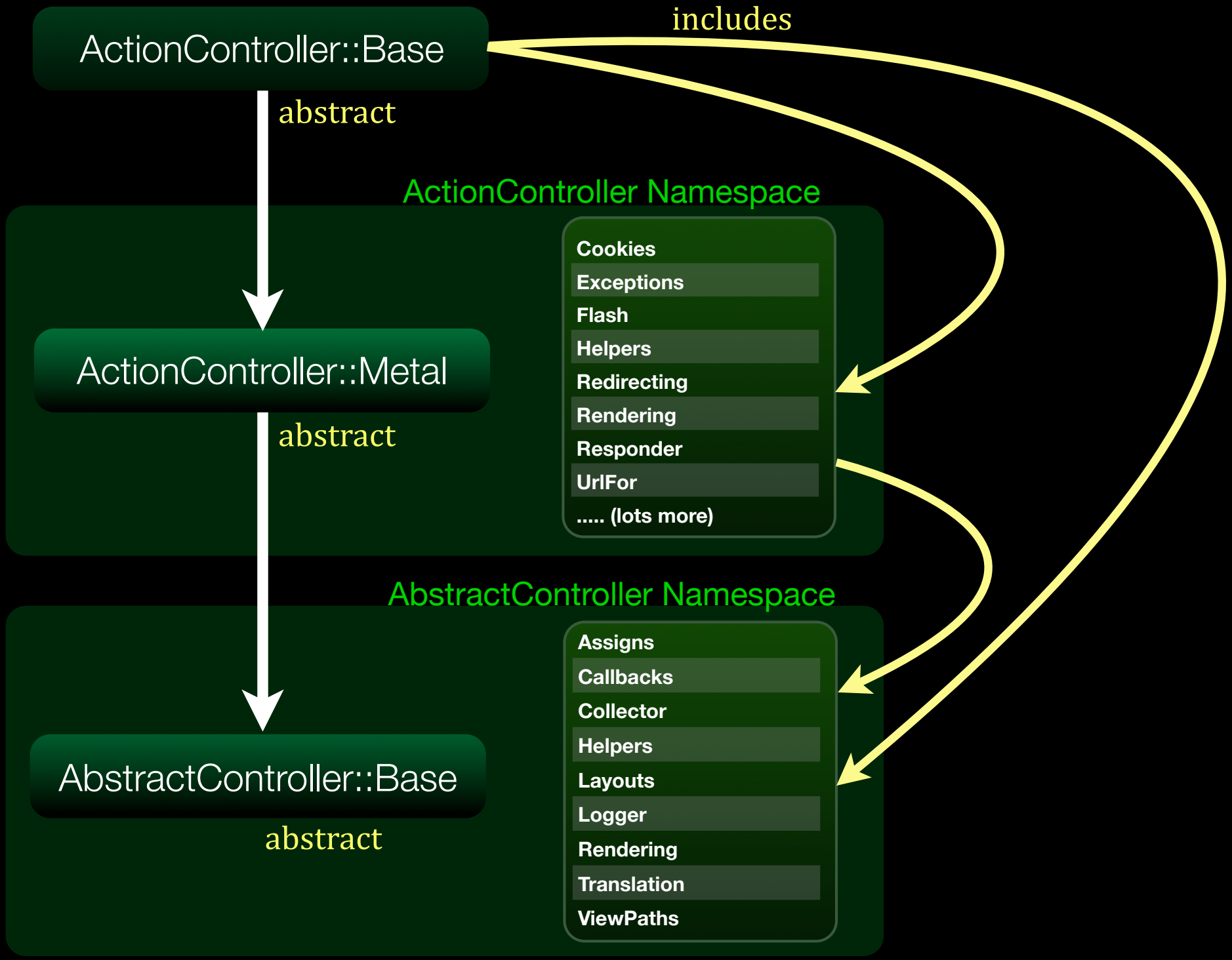
But I don’t want to rely on external servers for deployment




Bundler manages an **application's dependencies** through its entire life across many machines systematically and repeatably.

For more information

http://gembundler.com


ActionController::Base
includes
abstract
ActionController Namespace
Cookies
Exceptions
Flash
Helpers
Redirecting
Rendering
Responder
UrlFor
..... (lots more)
ActionController::Metal
abstract
AbstractController Namespace
Assigns
Callbacks
Collector
Helpers
Layouts
Logger
Rendering
Translation
ViewPaths
AbstractController::Base
abstract

# Regular Syntax

# New ActionController Syntax

```
class UsersController < ApplicationController

  def index
    @users = User.all

    respond_to do |format|
      format.html
      format.xml { render :xml => @users.to_xml }
    end
  end

  def show
    @user = User.find(params[:id])

    respond_to do |format|
      format.html # show.html.erb
      format.xml  { render :xml => @user }
    end
  end
...
```

# New ActionController Syntax

## Improved Syntax

```
class UsersController < ApplicationController
  respond_to :html, :xml, :json

  def index
    @users = User.all
    respond_with(@users)
  end

  def show
    @user = User.find(params[:id])
    respond_with(@user)
  end
...
```

# Tutorial - Lab #2

## Bundler & ActionController

# Rails 3 Ropes Course

- Getting Started & Routes
- Workshop - Lab #1
- Bundler & ActionController
- Workshop - Lab #2
- ActionMailer
- Workshop - Lab #3
- ActiveRelation & ActiveModel
- Workshop - Lab #4
- XSS & UJS
- Workshop - Lab #5

envylabs

Mikel Lindsaar

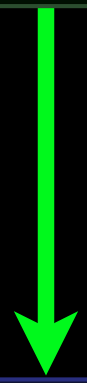
# ActionMailer

# New ActionMailer Syntax

Rails 2

```
$script/generate mailer UserMailer welcome forgot_password
```

```
create  app/models/user_mailer.rb
```



Rails 3

```
$rails g mailer UserMailer welcome forgot_password
```

```
create  app/mailers/user_mailer.rb
```

# New ActionMailer Syntax

**Rails 2**

```
def welcome(user, subdomain)
  subject 'Welcome to TestApp'
  recipients user.email
  from 'admin@testapp.com'

  body :user => user, :subdomain => subdomain
end
```

```
UserMailer.deliver_welcome(user, subdomain)
```

**Rails 3**

```
def welcome(user, subdomain)
  @user = user
  @subdomain = subdomain

  mail(:from => "admin@testapp.com",
       :to => user.email,
       :subject => "Welcome to TestApp")
end
```

```
UserMailer.welcome(user, subdomain).deliver
```

# New ActionMailer Syntax

**Rails 3**

```ruby
class UserMailer < ActionMailer::Base

  default :from => "admin@testapp.com",
          :reply_to => "noreply@testapp.com",
          "X-Time-Code" => Time.now.to_i.to_s

  def welcome(user, subdomain)
    @user = user
    @subdomain = subdomain

    attachments['test.pdf'] = File.read("#{Rails.root}/public/test.pdf")

    mail(:to => @user.email, :subject => "Welcome to TestApp") do |format|
      format.html { render 'other_html_welcome' }
      format.text { render 'other_text_welcome' }
    end
  end

end
```

## Defaults

welcome.text.erb
welcome.html.erb

# Tutorial - Step #3

## ActionMailer

# Rails 3 Ropes Course

- Getting Started & Routes
  - Workshop - Lab #1
- Bundler & ActionController
  - Workshop - Lab #2
- ActionMailer
  - Workshop - Lab #3
- **ActiveRelation & ActiveModel**
  - Workshop - Lab #4
- **XSS & UJS**
  - Workshop - Lab #5

envylabs

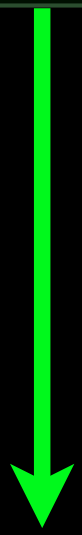
Nick Kallen

# ActiveRelation

replaces the internal ad-hoc query generation with query generation based on relational algebra.

Rails 2

```ruby
@posts = Post.find(:all, :conditions => {:published => true})
```



immediately queries the db

returns an Array of Posts

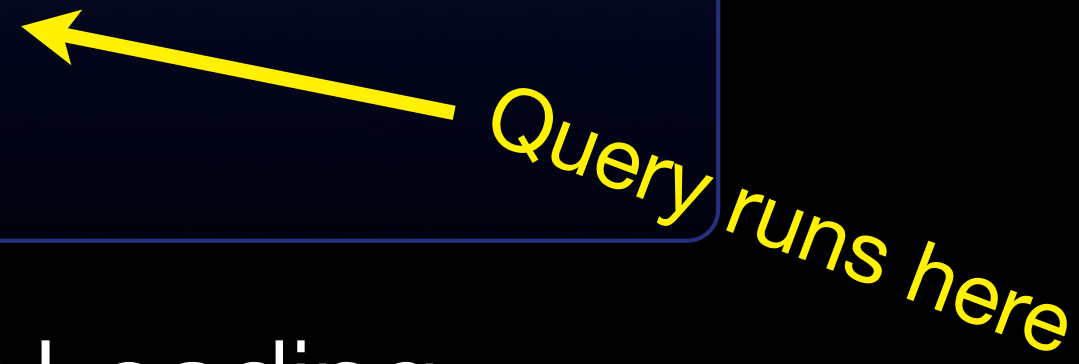
Rails 3

```ruby
@posts = Post.where(:published => true)
```

doesn't query the db

returns an ActiveRecord::Relation

```ruby
@posts = Post.where(:published => true)

if params[:order]
  @posts = @posts.order(params[:order])
end

@posts.each do |p|
  ...
end
```




# Lazy Loading

# ActiveRelation

```ruby
@posts = Post.where(:published => true)

if params[:order]
  @posts = @posts.order(params[:order])
end
```

```ruby
@posts = Post.where(:published => true)

@posts = @posts.order(params[:order])
```

```ruby
@posts = Post.where(:published => true).order(params[:order])
```

# ActiveRelation

```ruby
@posts = Post.where(:published => true).order(params[:order])
```

```ruby
posts = Post.order(params[:order])

@published = posts.where(:published => true)
@unpublished = posts.where(:published => false)
```

This is obviously a bad example

```ruby
@published = Post.published
@unpublished = Post.unpublished
```

# ActiveRelation

```ruby
@published = Post.published
@unpublished = Post.unpublished
```

**Rails 2**

```ruby
class Post < ActiveRecord::Base
  default_scope :order => 'title'

  named_scope :published, :conditions => {:published => true}
  named_scope :unpublished, :conditions => {:published => false}
end
```

**Rails 3**

```ruby
class Post < ActiveRecord::Base
  default_scope order('title')

  scope :published, where(:published => true)
  scope :unpublished, where(:published => false)
end
```

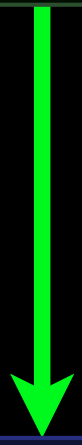
## New Finder Methods

```
where(:conditions)
having(:conditions)
select
group
order
limit
offset
joins
includes(:include)
lock
readonly
from
```

# ActiveRelation

**Rails 2**

```
Post.find(:all, :conditions => {:author => "Joe"}, :includes => :comments,
          :order => "title", :limit => 10)
```



**Rails 3**

```
Post.where(:author => "Joe").include(:comments).order(:title).limit(10).all
```

# ActiveModel

ActiveModel API

ActiveModel Helper Modules

## Old ActiveRecord Stack

Associations

Callbacks

Calculations

Dirty

Serialization

Migrations

Named Scope

Schema

Fixtures

Transactions

## ActiveModel

**Attribute Methods**

**Callbacks**

**Dirty**

**Errors**

**Naming**

**Observing**

**Serialization**

**Translation**

**Validations**

# ActiveModel

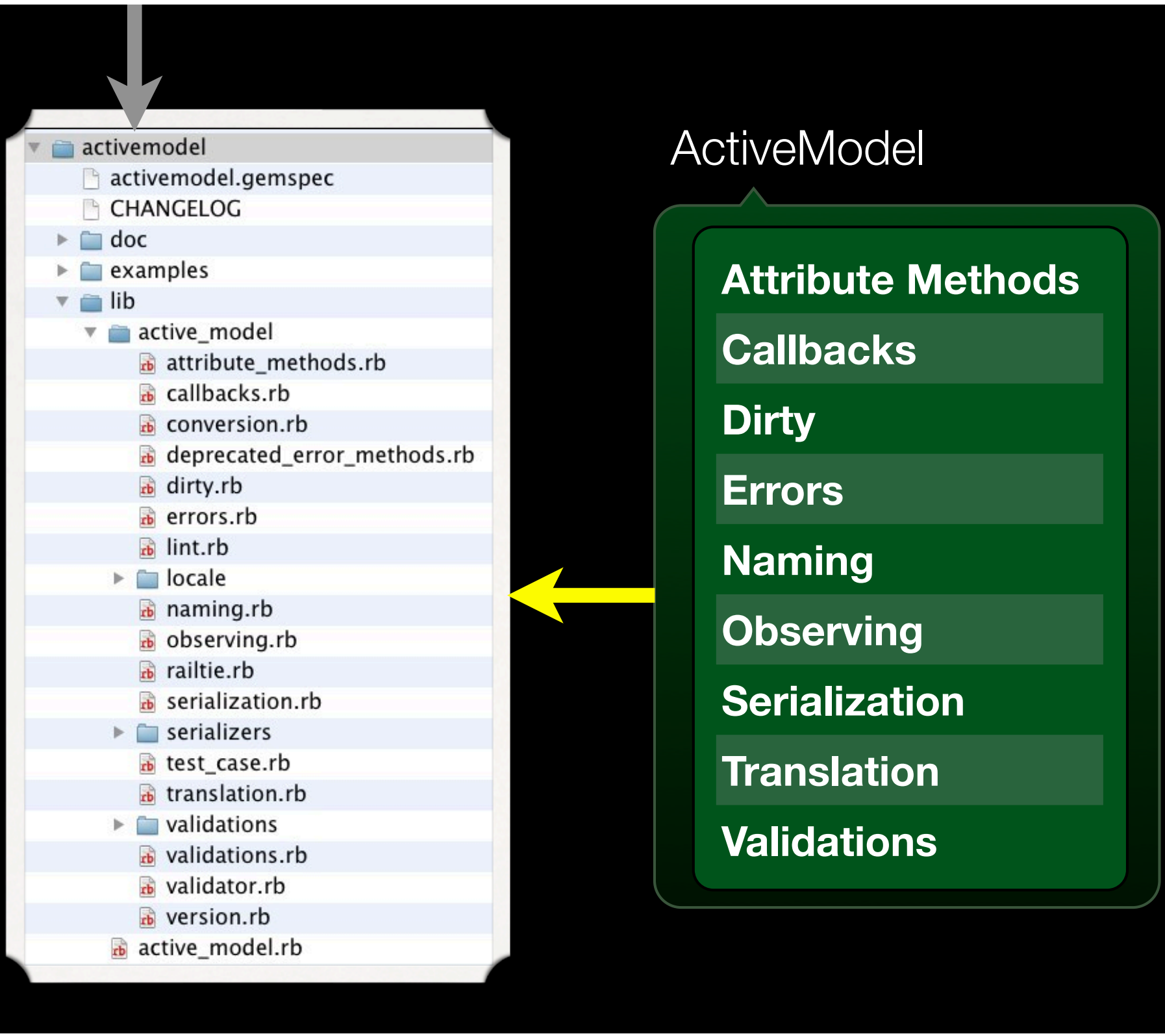
```
before_create :authenticate
before_save :send_email
around_create :log
```

```
person.changed?
person.name_changed?
person.name_was
person.name_change
```

```
person = Person.new
person.serializable_hash
person.to_json
person.to_xml
```

- **Attribute Methods**
- **Callbacks**
- **Dirty**
- **Errors**
- **Naming**
- **Observing**
- **Serialization**
- **Translation**
- **Validations**

```
validates_presence_of :email
validates_length_of :name, :within => 3..20
validates_inclusion_of :salary, :in => 50000..200000
```


activemodel
activemodel.gemspec
CHANGELOG
doc
examples
lib
active_model
attribute_methods.rb
callbacks.rb
conversion.rb
deprecated_error_methods.rb
dirty.rb
errors.rb
lint.rb
locale
naming.rb
observing.rb
railtie.rb
serialization.rb
serializers
test_case.rb
translation.rb
validations
validations.rb
validator.rb
version.rb
active_model.rb
ActiveModel
Attribute Methods
Callbacks
Dirty
Errors
Naming
Observing
Serialization
Translation
Validations

# Validations without db

```ruby
class Applicant
  include ActiveModel::Validations
  validates_presence_of :name, :email

  attr_accessor :name, :email
end
```

```
>> a = Applicant.new
 => #<Applicant:0x000001021b7198>
>> a.name = "Gregg"
 => "Gregg"
>> a.valid?
 => false
>> a.errors
 => {:email=>["can't be blank"]}
>> a.email = "Gregg@EnvyLabs.com"
 => "Gregg@EnvyLabs.com"
>> a.valid?
 => true
```

# Serialization without db

```ruby
class Applicant
  include ActiveModel::Serializers::JSON

  attr_accessor :name

  def attributes
    @attributes ||= {:name => 'nil'}
  end
end
```

```
>> a = Applicant.new
 => #<Applicant:0x00000102186d68>
>> a.name = "Gregg"
 => "Gregg"
>> a.to_json
 => "{\"name\":\"Gregg\"}"
```

# Tutorial - Lab #4

## ActiveRelation & ActiveModel

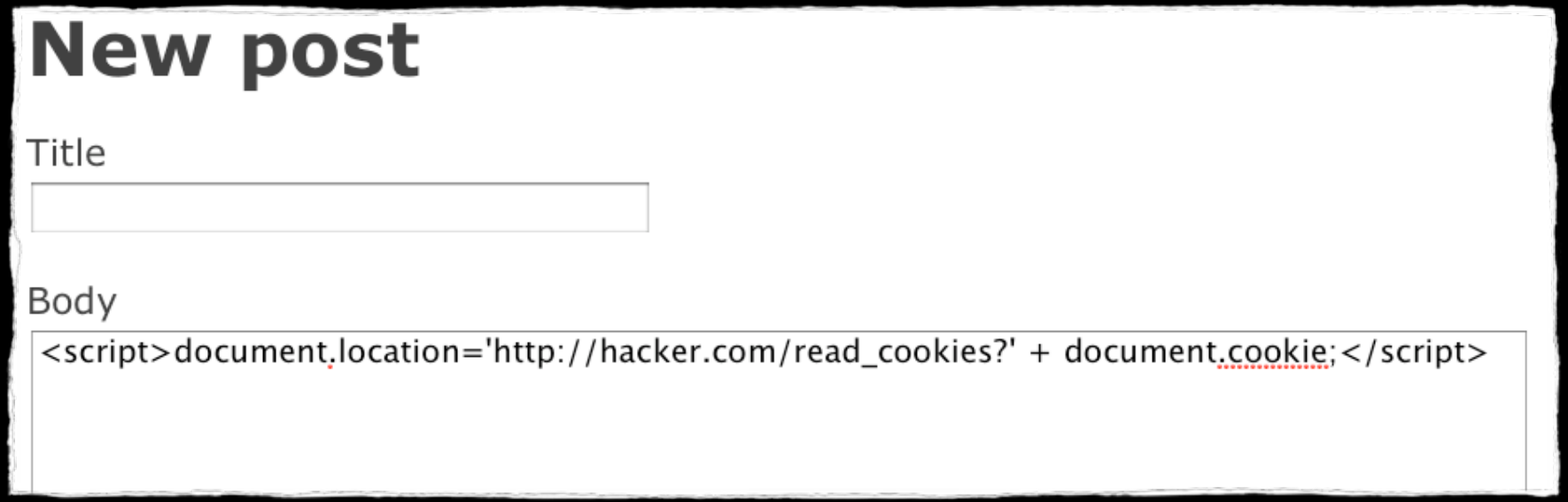
# Rails 3 Ropes Course

- **Getting Started & Routes**
  - Workshop - Lab #1
- **Bundler & ActionController**
  - Workshop - Lab #2
- **ActionMailer**
  - Workshop - Lab #3
- **ActiveRelation & ActiveModel**
  - Workshop - Lab #4
- **XSS & UJS**
  - Workshop - Lab #5

envylabs


# Cross Site Scripting

# Cross-Site Scripting (XSS)




**Rails 2**

```
<%= @post.body %>
```

(unsafe)

**Rails 2**

```
<%= h @post.body %>
```

(safe)

**Rails 3**

```
<%= raw @post.body %>
```

(unsafe)

**Rails 3**

```
<%= @post.body %>
```

(safe)

```
<%= raw @post.body %>
```

(unsafe)

```
<%= @post.body.html_safe %>
```

(unsafe)

action_view/helpers/raw_output_helper.rb

```
module ActionView
  module Helpers
    module RawOutputHelper
      def raw(stringish)
        stringish.to_s.html_safe
      end
    end
  end
end
```

```
>> s = "<h1>Yo!</h1>"
 => "<h1>Yo!</h1>"
>> s.html_safe?
 => false
>> s = s.html_safe
 => "<h1>Yo!</h1>"
>> s.html_safe?
 => true
```

Returns a String which is assumed to be safe and will not be escaped again

# Cross-Site Scripting (XSS)

**Rails 2**

```
<%= link_to "<span class='cart'>Cart</span>", cart_path %>
```

**Rails 3**

```
<%= link_to raw("<span class='cart'>Cart</span>"), cart_path %>
```

**Rails 3**

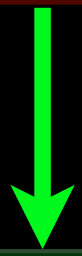
```
<%= link_to raw("<span class='cart'>#{@user_input}</span>"), cart_path %>
```

*Vulnerable to attack!!*

*No escaping done on user input*

# Cross-Site Scripting (XSS)

```
<%= link_to raw("<span class='cart'>#{@user_input}</span>"), cart_path %>
```



```
<%= link_to raw("<span class='cart'>#{h @user_input}</span>"), cart_path %>
```

# UJS Unobtrusive Javascript

# Adopting Unobtrusive Javascript

## HTML 5 custom data attributes `data-*`

Custom data attributes are intended to store custom data private to the page or application, for which there are no more appropriate attributes or elements

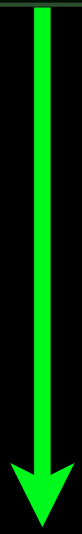
- `data-remote`
- `data-method`
- `data-confirm`
- `data-disable-with`

# Adopting Unobtrusive Javascript

**Rails 2**

```
<%= link_to_remote 'Show', :url => post %>
```

```
<a href="#" onclick="new Ajax.Request('/posts/1', {asynchronous:true,
evalScripts:true, parameters:'authenticity_token=' +
encodeURIComponent('9sk..44d')}); return false;">Show</a>
```



**Rails 3**

```
<%= link_to 'Show', post, :remote => true %>
```

```
<a href="/posts/1" data-remote="true">Show</a>
```

# Adopting Unobtrusive Javascript

**Rails 2**

```
<% remote_form_for(@post) do |f| %>
```

```
<form action="/posts" class="new_post" id="new_post" method="post"
onsubmit="new Ajax.Request('/posts', {asynchronous:true,
evalScripts:true, parameters:Form.serialize(this)}); return false;">
```

**Rails 3**

```
<%= form_for(@post, :remote => true) do |f| %>
```

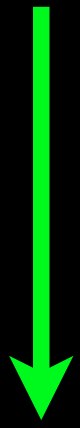
```
<form action="/posts" class="new_post" data-remote="true" id="new_post" method="post">
```

# Adopting Unobtrusive Javascript

**Rails 2** **Rails 3**

```
<%= link_to 'Destroy', post, :method => :delete %>
```

```
<a href="/posts/1" onclick="var f = document.createElement('form'); f.style.display = 'none';
this.parentNode.appendChild(f); f.method = 'POST'; f.action = this.href;var m = document.createElement
('input'); m.setAttribute('type', 'hidden'); m.setAttribute('name', '_method'); m.setAttribute('value',
'delete'); f.appendChild(m);var s = document.createElement('input'); s.setAttribute('type', 'hidden');
s.setAttribute('name', 'authenticity_token'); s.setAttribute('value', '9skdJ0k+l9/
q3PWToz6MtfyiB2gcyhnKubeGV6WFL44='); f.appendChild(s);f.submit();return false;">Destroy</a>
```



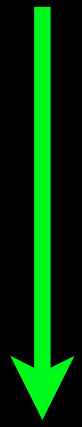
```
<a href="/posts/1" data-method="delete" rel="nofollow">Destroy</a>
```

# Adopting Unobtrusive Javascript

**Rails 2** **Rails 3**

```
<%= link_to 'Destroy', post, :confirm => 'Are you sure?', :method => :delete %>
```

```
<a href="/posts/1" onclick="if (confirm('Are you sure?')) { var f = document.createElement('form');
f.style.display = 'none'; this.parentNode.appendChild(f); f.method = 'POST'; f.action = this.href;var m =
document.createElement('input'); m.setAttribute('type', 'hidden'); m.setAttribute('name', '_method');
m.setAttribute('value', 'delete'); f.appendChild(m);var s = document.createElement('input'); s.setAttribute
('type', 'hidden'); s.setAttribute('name', 'authenticity_token'); s.setAttribute('value', '9skdJ0k+l9/
q3PWToz6MtfyiB2gcyhnKubeGV6WFL44='); f.appendChild(s);f.submit(); };return false;">Destroy</a>
```

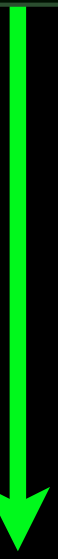


```
<a href="/posts/1" data-confirm="Are you sure?" data-method="delete" rel="nofollow">Destroy</a>
```

# Adopting Unobtrusive Javascript

**Rails 2**

```
<%= f.submit 'Create Post', :disable_with => "Please wait..." %>
```



```
<input id="post_submit" name="commit" onclick="if (window.hiddenCommit)
{ window.hiddenCommit.setAttribute('value', this.value); }else { hiddenCommit =
document.createElement('input');hiddenCommit.type = 'hidden';hiddenCommit.value =
this.value;hiddenCommit.name = this.name;this.form.appendChild(hiddenCommit); }
this.setAttribute('originalValue', this.value);this.disabled = true;this.value='Please
wait...';result = (this.form.onsubmit ? (this.form.onsubmit() ? this.form.submit() :
false) : this.form.submit());if (result == false) { this.value = this.getAttribute
('originalValue');this.disabled = false; }return result;" type="submit" value="Create
Post" />
```

**Rails 3**

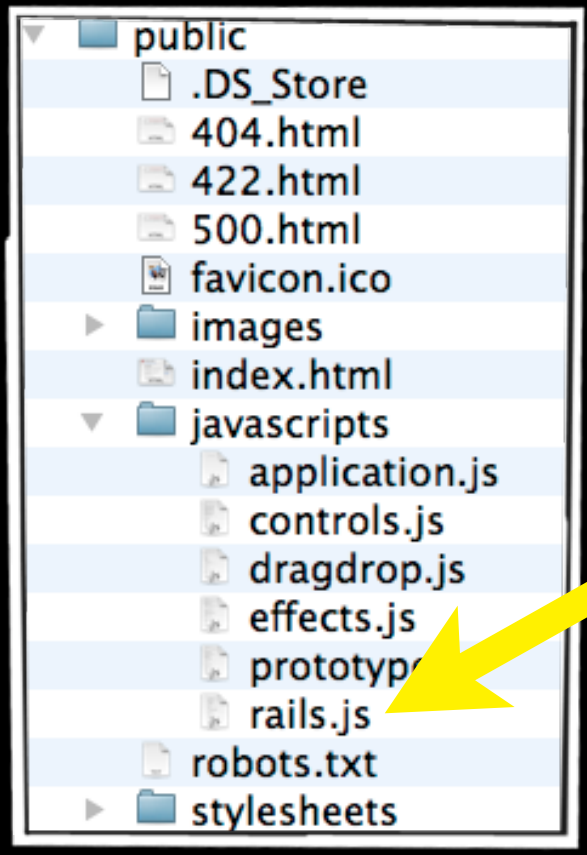
```
<%= f.submit :disable_with => "Please wait..." %>
```

```
<input data-disable-with="Please wait..."
                id="post_submit" name="commit" type="submit" value="Create Post" />
```


# Adopting Unobtrusive Javascript

## HTML 5 custom data attributes

- `data-remote`
- `data-method`
- `data-confirm`
- `data-disable-with`



# /public/javascripts/rails.js

```
document.observe("dom:loaded", function() {

  $(document.body).observe("click", function(event) {

    var message = event.element().readAttribute('data-confirm');
    if (message) {
      // ... Do a confirm box
    }

    var element = event.findElement("a[data-remote]");
    if (element) {
      // ... Do the AJAX call
    }

    var element = event.findElement("a[data-method]");
    if (element) {
      // ... Create a form
    }
  });
```

# jQuery in Rails?

http://github.com/rails/jquery-ujs

**jquery-ujs** / src

| name | age | message | history |
|---|---|---|---|
| .. | | | |
| rails.js | February 06, 2010 | Get rid of trailing whitespace [foca] | |

```
$('a[data-confirm],input[data-confirm]').live('click', function () {
   // ... Do a confirm box
});

$('form[data-remote]').live('submit', function (e) {
   // ... Do an AJAX call
});
```

# Deprecated Methods

```
link_to_remote
remote_form_for
observe_field
observe_form
form_remote_tag
button_to_remote
submit_to_remote
link_to_function
periodically_call_remote
```

**prototype_legacy_helper**

http://github.com/rails/prototype_legacy_helper

# Cross-site Request Forgery

## Hacker site

```
<form target="http://yoursite.com">
```

→ Your site

**Rails 2** **Rails 3**

```
class ApplicationController < ActionController::Base
  protect_from_forgery
end
```

**Rails 2**

```
<% form_for(@post) do |f| %>
```

```
<input name="authenticity_token" type="hidden" value="vg..4=" />
```

**Rails 3**

```
<%= csrf_meta_tag %>
```

*(in your layout)*

```
<meta name="csrf-param" content="authenticity_token"/>
<meta name="csrf-token" content="I+d..jI="/>
```

rails.js unobtrusively adds the token

# Tutorial - Lab #5

## XSS & UJS

# Rails 3 Ropes Course

**Getting Started & Routes**

Workshop - Lab #1

**Bundler & ActionController**

Workshop - Lab #2

**ActionMailer**

Workshop - Lab #3

**ActiveRelation & ActiveModel**

Workshop - Lab #4

**XSS & UJS**

Workshop - Lab #5

envylabs

# Internal APIs

(that we didn't cover)

**Generators**

**ActiveModel API**

**Custom Validations**

**ActionController Modularity**

**Railties**

# Creative Commons

| name | author | URL |
|---|---|---|
| rainbow of 80s toys | merwing✿little dear | http://www.flickr.com/photos/merwing/2152164258/ |
| Notting Hill Gate | Eole | http://www.flickr.com/photos/eole/942309733/ |
| Das Licht | Small | http://www.flickr.com/photos/small/62713023/ |
| Metro Genova | opti mystic | http://www.flickr.com/photos/miiilio/2503634282/ |
| Immobility Dilemna | gilderic | http://www.flickr.com/photos/gilderic/3528157964/ |
| train station | nolifebeforecoffee | http://www.flickr.com/photos/nolifebeforecoffee/1803584805/ |
| Mystical station | Jsome1 | http://www.flickr.com/photos/jsome1/2226394415/ |
| Railswaystation | Pieter Musterd | http://www.flickr.com/photos/piet_musterd/2233025691/ |
| The Handover | MarkyBon | http://www.flickr.com/photos/markybon/152769885/ |
| EN57 | magro_kr | http://www.flickr.com/photos/iks_berto/1328682171/ |
| Adirondack Extreme | Mikey Roach | http://www.flickr.com/photos/mikeroach/4576888456/ |
| IMG_1242 | khoogheem | http://www.flickr.com/photos/khoogheem/3534078991/ |

***Presentation by:***


*http://envylabs.com*

Ruby5 Podcast
*http://ruby5.envylabs.com*

***If you need help with a Rails 3 project, feel free to give us a call***

**Gregg Pollack**
*Gregg@EnvyLabs.com*

**Nathaniel Bibler**
*Nate@EnvyLabs.com*

**Thomas Meeks**
*Thomas@EnvyLabs.com*

**Jacob Swanner**
*Jacob@EnvyLabs.com*

**Tyler Hunt**
*Tyler@EnvyLabs.com*

**Mark Kendall**
*Mark@EnvyLabs.com*

**Caike Souza**
*Caike@EnvyLabs.com*