

Michael Mahlberg, Consulting Guild AG
Jens-Christian Fischer, InVisible GmbH

SOLID Ruby - SOLID Rails

Establishing a sustainable
codebase

The background features a stylized, colorful train moving from the bottom left towards the top right. The train is composed of various layers of curved lines in shades of orange, yellow, and white. It leaves a trail of similar colored lines behind it. The train is positioned on a set of white, curved railway tracks that curve upwards and to the right. The overall background is a dark grey with subtle diagonal lines.

RailsWayCon

A photograph of a street scene. In the center, there's a tan-colored building with a white-framed window. A white sign is mounted on the building, with the words "OLD THINGS" printed on it in a serif font. To the left of the building, there's a dark wooden utility pole. In the background, there are more buildings and some trees under a clear blue sky.

Who?

OLD
THINGS

Michael Mahlberg

RailsWayCon

Founder of

roughly a dozen companies
over the last two decades

RailsWayCon

>> relevance

=> nil

Working as

A consultant on software
processes, architecture &
design for > 2 decades

RailsWayCon

```
>> relevance != nil  
=> true
```



Who?

Jens-Christian Fischer

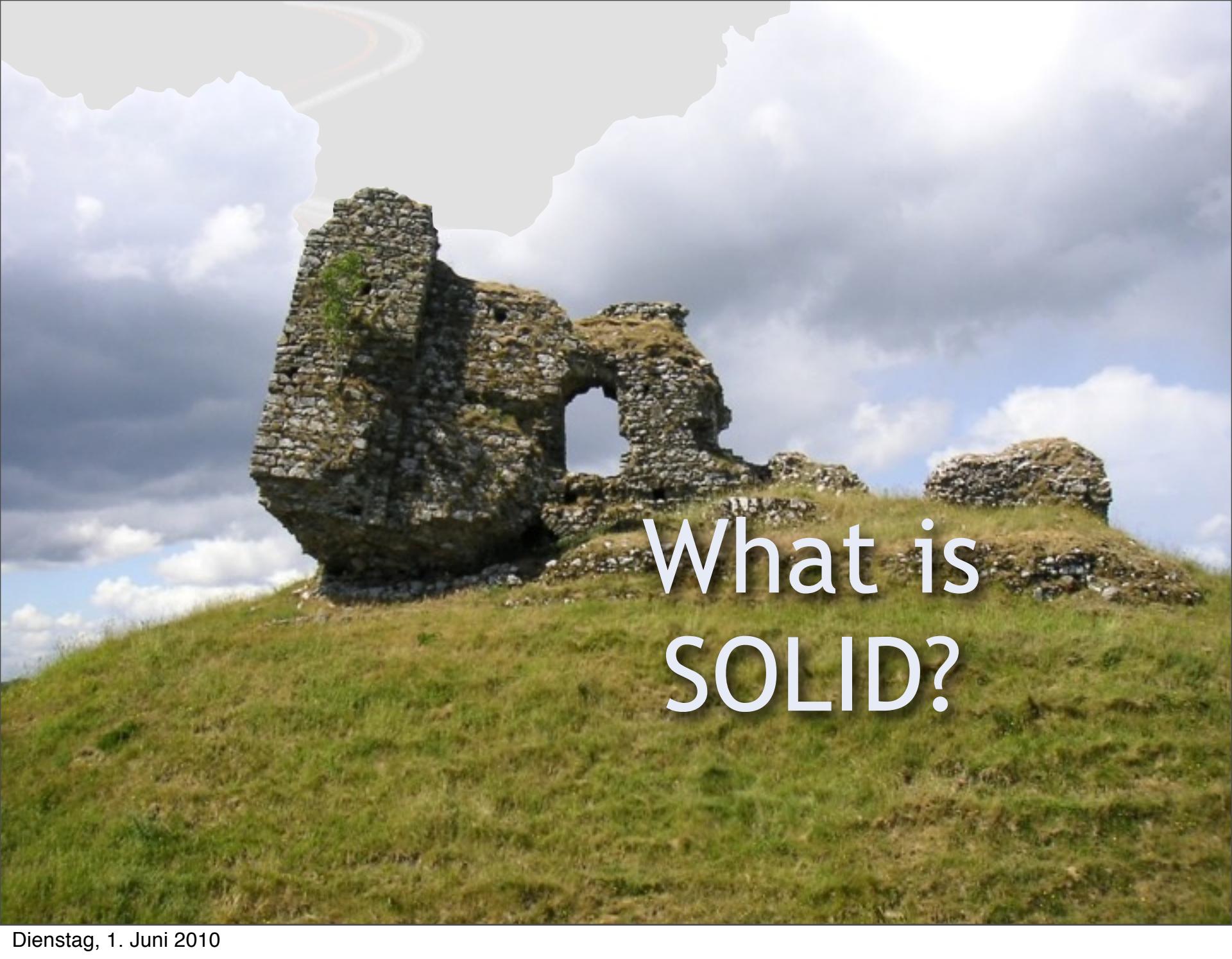
RailsWayCon

Dienstag, 1. Juni 2010

Tinkerer, Practician, Author

and generally
interested in way
too many things

RailsWayCon

The background image shows a large, crumbled stone castle ruin situated on a grassy hill. The sky above is filled with various shades of grey and white clouds. In the center of the image, the words "What is SOLID?" are written in a large, bold, white sans-serif font.

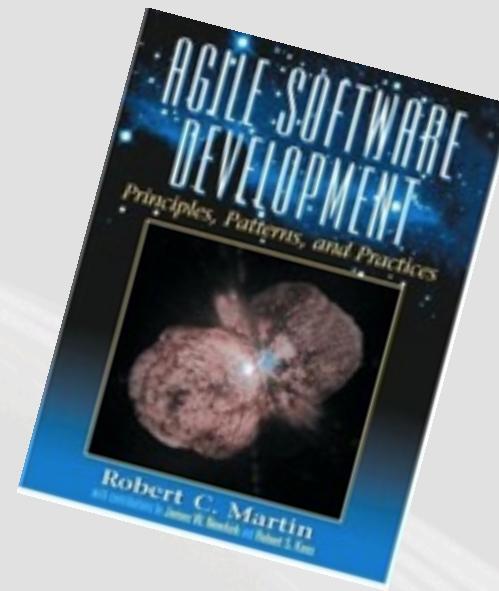
What is
SOLID?



SOLID
is
not
a
Law

PPP

(by Robert C. Martin)



Agile Software
Development,
Principles, Patterns,
and Practices

RailsWayCon

Principles!

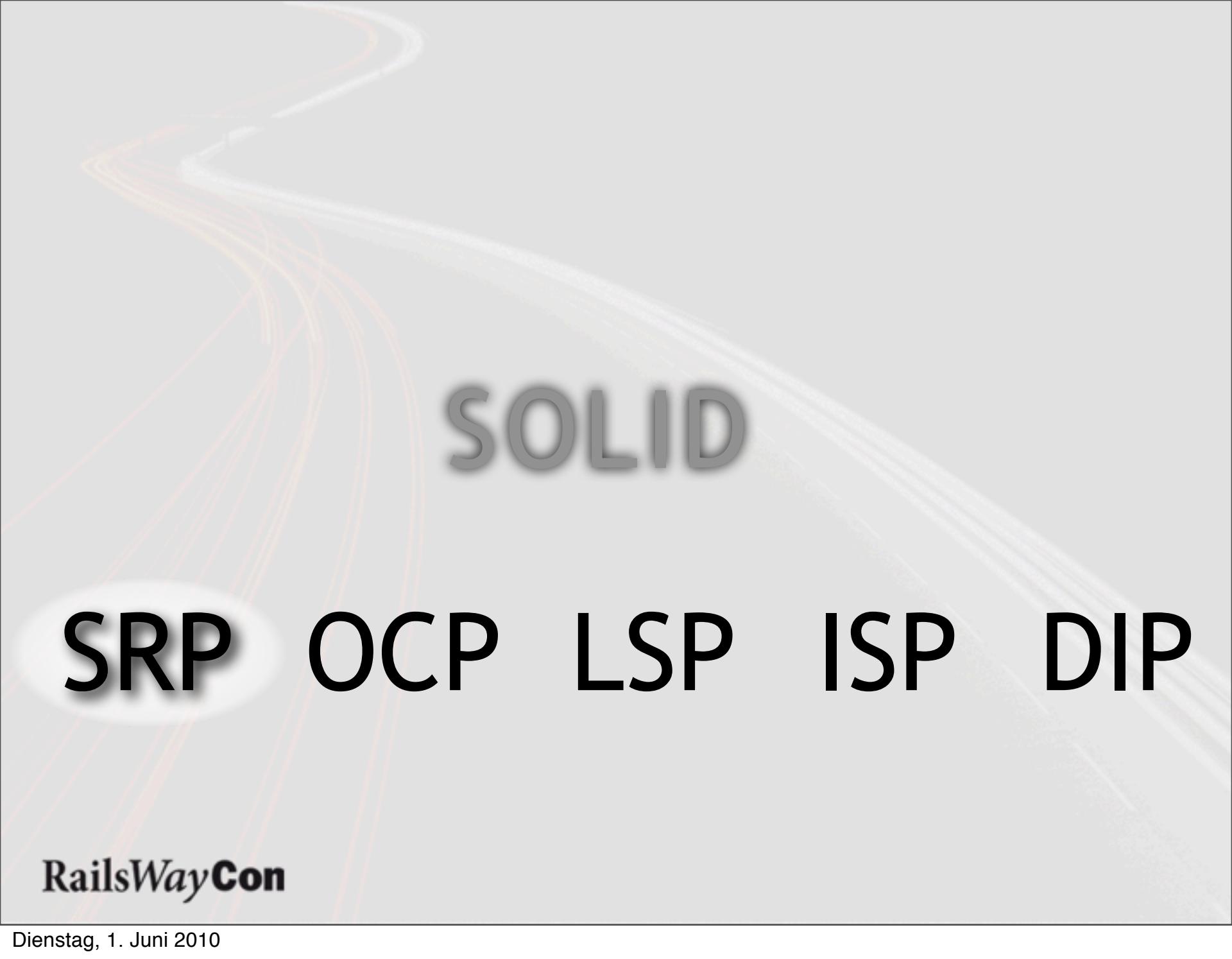
You know - more
like guidelines

DANGER
GEFAHR



SRI OCP LSP ISP DIP

RailsWayCon



SOLID

SRP OCP LSP ISP DIP

RailsWayCon

SRP

Single Responsibility Principle



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

RailsWayCon

```
require 'digest/sha1'

class User < ActiveRecord::Base
  include Authentication
  include Authentication::ByPassword
  include Authentication::ByCookieToken

  #TODO Check login redirect if this filter is skipped
  #skip_after_filter :store_location

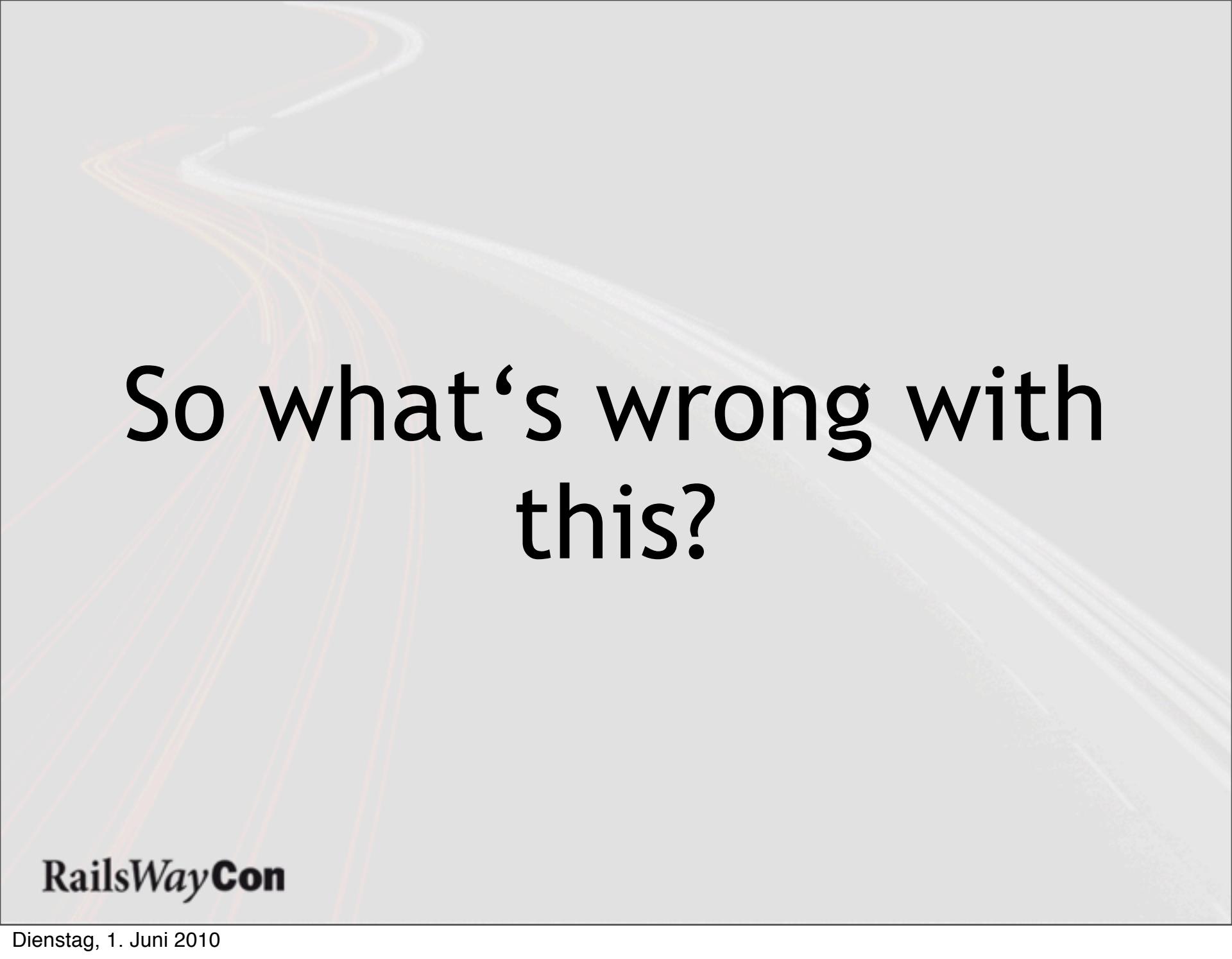
  # Virtual attribute for the unencrypted password
  attr_accessor :password

  belongs_to :country

  has_one :user_profile, :dependent => :destroy
  has_one :note

  has_many :queries
  has_many :tags, :foreign_key => "created_by"
  has_many :taggings, :as => :tagger
  has_many :organizations, :through => :affiliations
  has_many :affiliations
  has_many :locations, :through => :affiliations
  has_many :projects, :through => :memberships
  has_many :memberships
  has_many :public_assets, :through => :privileges
  has_many :descriptions, :through => :privileges
  has_many :assessments, :through => :privileges
  has_many :description_profiles, :through => :privileges
  has_many :privileges
  has_many :diaries
  has_many :roles, :through => :commitments
  has_many :commitments
  has_many :activities
  has_many :messages
  has_many :fellowships
  has_many :user_groups, :through => :fellowships
```

User Class



So what's wrong with this?

RailsWayCon

From: user.rb

```
class User < ActiveRecord::Base
  include Authentication
  include Authentication::ByPassword
  include Authentication::ByCookieToken
...
  belongs_to :country
...
  has_one :user_profile, :dependent => :destroy
  has_many :queries
  has_many :tags, :foreign_key => "created_by"
...
  validates_presence_of      :login, :email, :country_id
  validates_presence_of      :password, :if => :password_required?
...

```

From: user.rb

```
acts_as_state_machine :initial => :pending

state :pending, :enter => :make_activation_code
state :active, :enter => :do_activate
...
event :register do
  transitions :from => :passive, :to => :pending, :guard =>
Proc.new { |u| !(u.crypted_password.blank? &&
u.password.blank?) }
end
...
def message_threads
  self.message_threads + self.message_threads
end
```

From: user.rb

```
def forum_nickname
  self.user_profile.nickname.blank? ? "#{self.first_name} #
{self.last_name}" : self.user_profile.nickname
end

def name
  "#{self.first_name} #{self.last_name}" rescue 'n/a'
end

def email_with_name
  "#{self.first_name} #{self.last_name} <#{self.email}>"
end
```

From: user.rb

```
def is_admin?  
  self.roles.collect{|role| role.title}.include?('admin')  
end  
  
def countries  
  [self.country]  
end
```

From: user.rb

```
def boards
  Board.all :conditions => { :user_group_id =>
self.user_groups.collect{ |g| g.id } }
end

def discussions
  Discussion.all :conditions => { :board_id =>
self.boards.collect{ |b| b.id } }
end

def organization_roles
  role_ids = Affiliation.all(:conditions => { :user_id =>
self.id }).collect{|a| a.role_id}.uniq
  roles = Role.find(role_ids)
end
```

From: user.rb

```
def make_password_reset_code
  self.password_reset_code = Digest::SHA1hexdigest(
    Time.now.to_s.split("//").sort_by {rand}.join )
end

def self.published_users
  User.all(:conditions => ['state = ?',
  'published'], :order => 'login ASC', :include =>
  [:user_profile])
end
```

Anyone notice a pattern?

RailsWay**Con**



Neither do we

RailsWayCon

Separation of Concerns

RailsWay**Con**



Authentication Roles Mailers State

• • •

RailsWay**Con**

So how?

Mixins

RailsWay**Con**

New User Model

```
class User < ActiveRecord::Base
  include Authentication
  include Authentication::ByPassword
  include Authentication::ByCookieToken

  include Project::UserStates
  include Project::UserMailer
  include Project::UserForum
  include Project::UserMessages

...
end
```

UserMessages

```
module Project
  module UserMessages
    # to be included in User Model

    has_many :messages
    def message_threads
      MessageThread.all(:conditions =>
        ["sender_id = ? or receiver_id = ?",
         self.id, self.id])
    end
  end
end
```

Methods

RailsWay**Con**

```
def transfer(data, url)
  h = Net::HTTP.new(self.uri.host, self.uri.port)
  RAILS_DEFAULT_LOGGER.debug "connecting to CL: #{self.uri}"
  RAILS_DEFAULT_LOGGER.debug "connecting to CL: #{url}"

  resp = h.post(url, data, {'Content-Type' => 'application/xml'})
  response_code = resp.code.to_i
  location = if response_code == 201
    resp['Location']
  else
    RAILS_DEFAULT_LOGGER.debug "error from CL: #{response_code}"
    RAILS_DEFAULT_LOGGER.debug "error from CL: #{resp.body}"
    @error = resp.body
    nil
  end
  [response_code, location]
end
```

```
def transfer(data, document)
  if document.cl_document_url != nil
    self.uri = URI.parse(document.cl_document_url )
    h = Net::HTTP.new(self.uri.host, self.uri.port)
    response = h.post(self.uri, data, {'Content-Type' =>
'application/xml'})
  else
    h = Net::HTTP.new(self.uri.host, self.uri.port)
    response = h.post("/tasks", data, {'Content-Type' =>
'application/xml'})
  end
  response_code = response.code.to_i
  if response_code == 201
    location = response['Location']
    document.cl_document_url = location
    document.save!
  else
    nil
  end
  [response_code, location]
end
```

SRP Transfer

```
def transfer data
  open_connection
  post data
  return location
end

def open_connection
  @http = Net::HTTP.new(self.uri.host, self.uri.port)
end

def post data
  @response = http.post(self.url, data, {'Content-Type' =>
                                             'application/xml'})
end
```

```
def location
  get_location if created? # returns nil if not created?
end

def response_code
  @response.code.to_i
end

def created?
  response_code == 201
end

def get_location
  @response['Location']
end

def error
  @response.body
end
```

Add a 16-band
equalizer & a
BlueRay®
player to this...

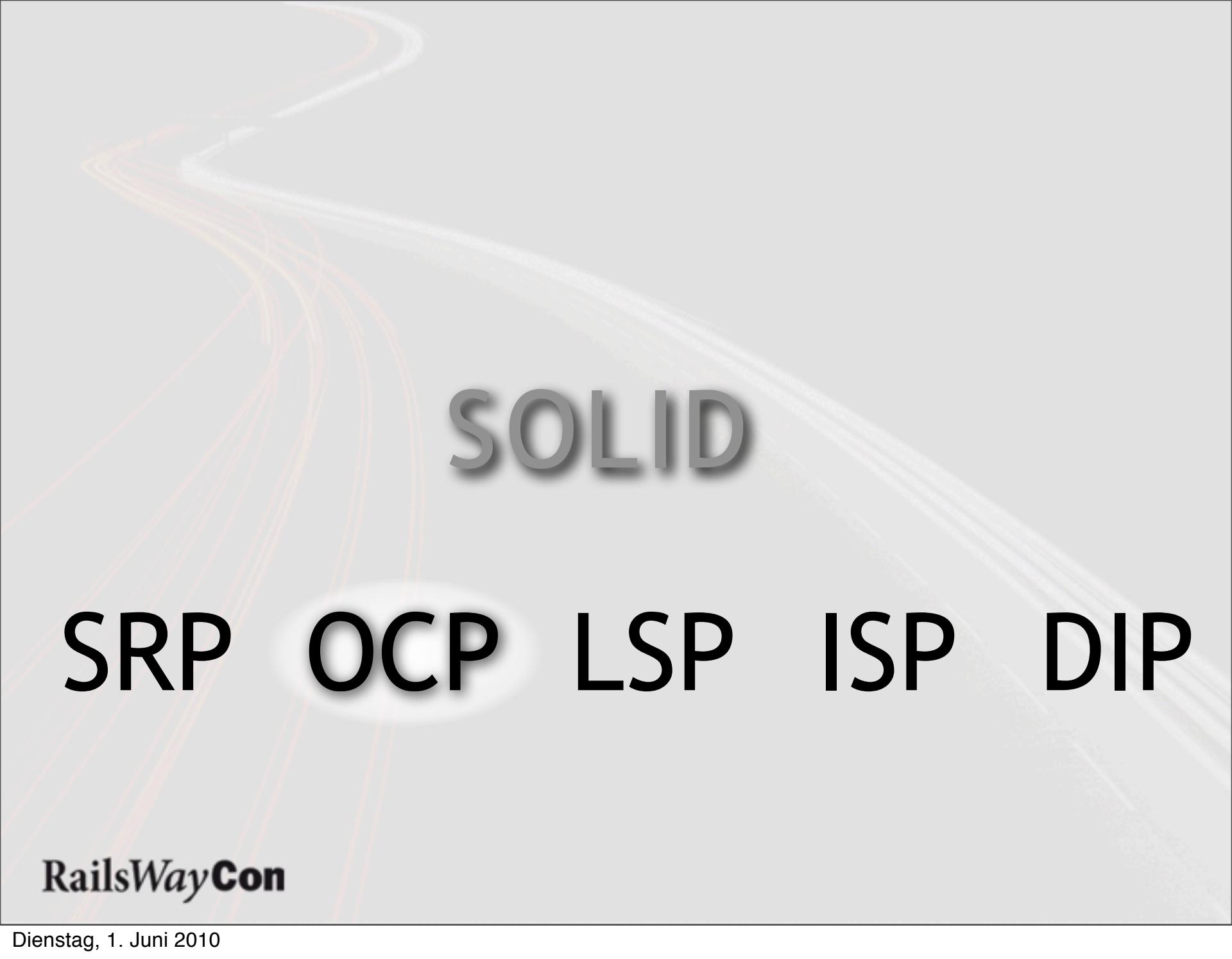


RailsWayCon

And now to
this...



RailsWay**Con**



SOLID

SRP OCP LSP ISP DIP

RailsWayCon

OCP

Open Closed Principle



You should be able
to extend a classes
behavior, without
modifying it.



RailsWay**Con**

Dienstag, 1. Juni 2010



RailsWay**Con**

Dienstag, 1. Juni 2010

Overview

[Home](#)

[RESULTS!](#)

[Schools](#)

[Countries](#)

Tron Materials

[Write Your Own Tron Bot](#)

[Forums](#)

[Articles & Tutorials](#)

[Frequently Asked Questions](#)

[Final Tournament](#)



Final Rankings Announced!

Congratulations to this year's Google AI Challenge Grand Champion, **Andy Sloane!** Andy is originally from Milwaukee, Wisconsin, and is currently working as a software engineer at Yahoo!. His winning entry was entitled `a1k0n_`. Congratulations!

Congratulations also go out to each of this year's finalists (top 100). Achieving such a high ranking takes a lot of work, and you should all be very proud!

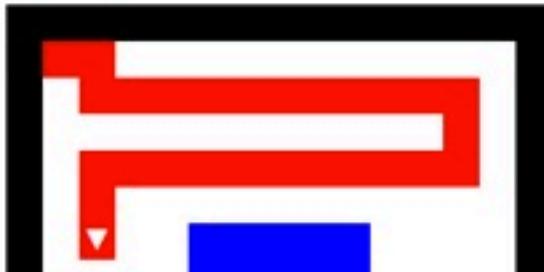
On behalf of Google and the University of Waterloo Computer Science Club, I would like to thank everybody who took part in the Google AI Challenge. This year was the best yet, with more than 1400 working entries submitted by people from 79 different countries. Thank you!

Full results can be found [here](#).

Tron Game Viewer

After the video from the last post instantly hit number 1 on [Reddit](#), it occurred to us that maybe people would like to be able to see instant replays of their own games. We are very proud to announce that you can now watch any game using the [Tron Game Viewer!](#)

To see your games, visit your profile, where you'll find a link next to each of your most recent games that leads to the Tron Game Viewer. You can even put a link to one of your games in your [Gchat](#), [Facebook](#), or [MSN status](#).



Overview

[Home](#)

[RESULTS!](#)

[Schools](#)

[Countries](#)

Tron Materials

[Write Your Own Tron Bot](#)

[Forums](#)

[Articles & Tutorials](#)

[Frequently Asked Questions](#)

[Final Tournament](#)



Ruby's User Rankings

[« Back to Main Leaderboard](#)

RANK	USERNAME	COUNTRY	ORGANIZATION	LANGUAGE	ELO SCORE
1 (89)	_geot_		Other	Ruby	2415
2 (101)	naoh87		Other	Ruby	2398
3 (146)	stungeye		Other	Ruby	2259
4 (154)	pcsi1Ruby		Other	Ruby	2240
5 (169)	jagoterr		Other	Ruby	2201
6 (188)	samwilkeson		Other	Ruby	2141
7 (217)	bvcbvc		Other	Ruby	2103
8 (263)	buddhabrot		Other	Ruby	2038
9 (263)	gspiers		Other	Ruby	2038
10 (283)	tingar		Portland State	Ruby	2009
11 (315)	abbeyandzoe		Other	Ruby	1980
12 (319)	mattkobelski		McGill University	Ruby	1977
13 (321)	jcfischer		Other	Ruby	1976
14 (357)	benofsky		Gonzaga College S.J.	Ruby	1952
15 (398)	bobaman		Google Inc	Ruby	1914
16 (442)	will.bailey		Other	Ruby	1873
17 (459)	CreativeEmbassy		Other	Ruby	1858
18 (478)	context		Other	Ruby	1832
19 (484)	tumbleweed		Other	Ruby	1826
20 (519)	adamgamble		Other	Ruby	1794
21 (583)	zeotrope		University of Waterloo	Ruby	1717

From the Google AI Challenge (Tronbot)

```
def makemove(map)
  x, y = map.my_position
  # calculate a move ...
  if(valid_moves.size == 0)
    map.make_move( :NORTH )
  else
    # choose move ...
    puts move # debug (like in the old days)
    map.make_move( move )
  end
end

class Map
  ...
  def make_move(direction)
    $stdout << ({:NORTH=>1, :SOUTH=>3, :EAST=>2, :WEST=>4}[direction])
    $stdout << "\n"
    $stdout.flush
  end
end
```

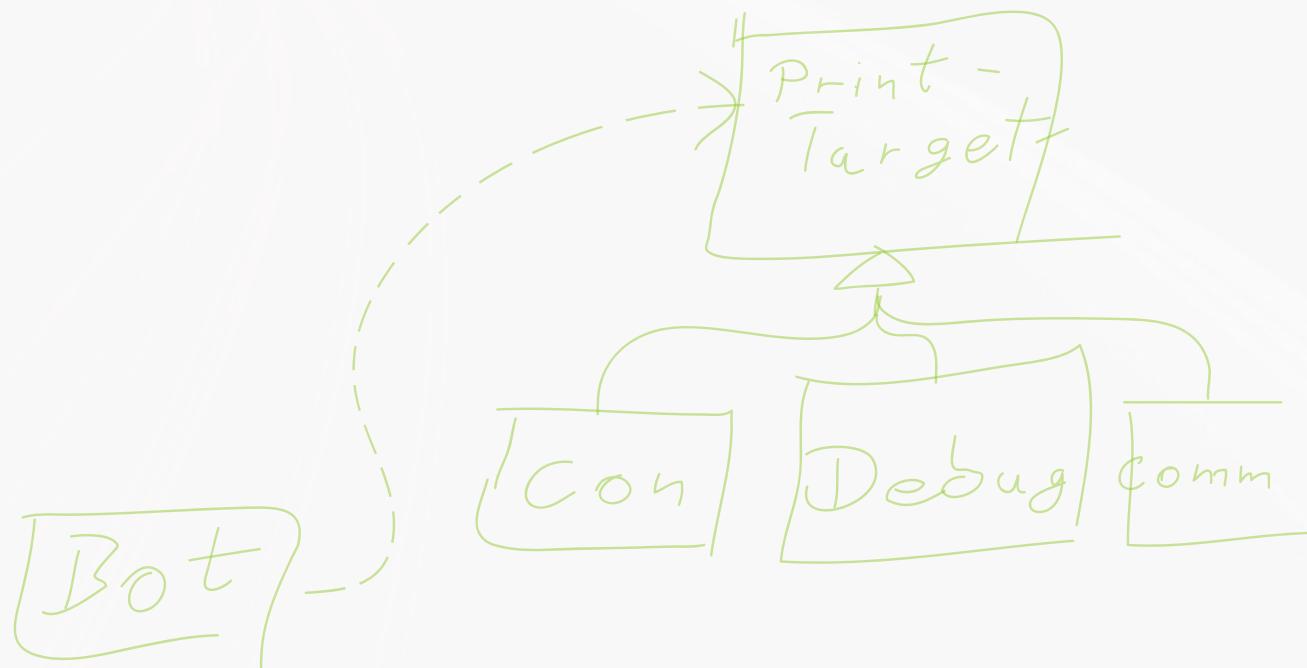
From the Google AI Challenge (Tronbot)

```
def puts(*args)
  $stderr.puts *args
end

def p(*args)
  args.map!{|arg| arg.inspect}
  puts args
end

def print(*args)
  $stderr.print *args
end
```

Design Sketch



```
class Outputer

  def initialize(io = $stderr)
    @io = io
  end

  def puts(*args)
    @io.puts *args
  end

  ...
end

out = Outputer.new
out.puts "Testing"
```



SOLID

SRP OCP LSP ISP DIP

RailsWay**Con**

LSP

Liskov Substitution Principle



Derived classes
must be substitutable
for their base
classes.

RailsWayCon

No Problem in Ruby

Or so it seems...

RailsWay**Con**

No Interface...

no problem?

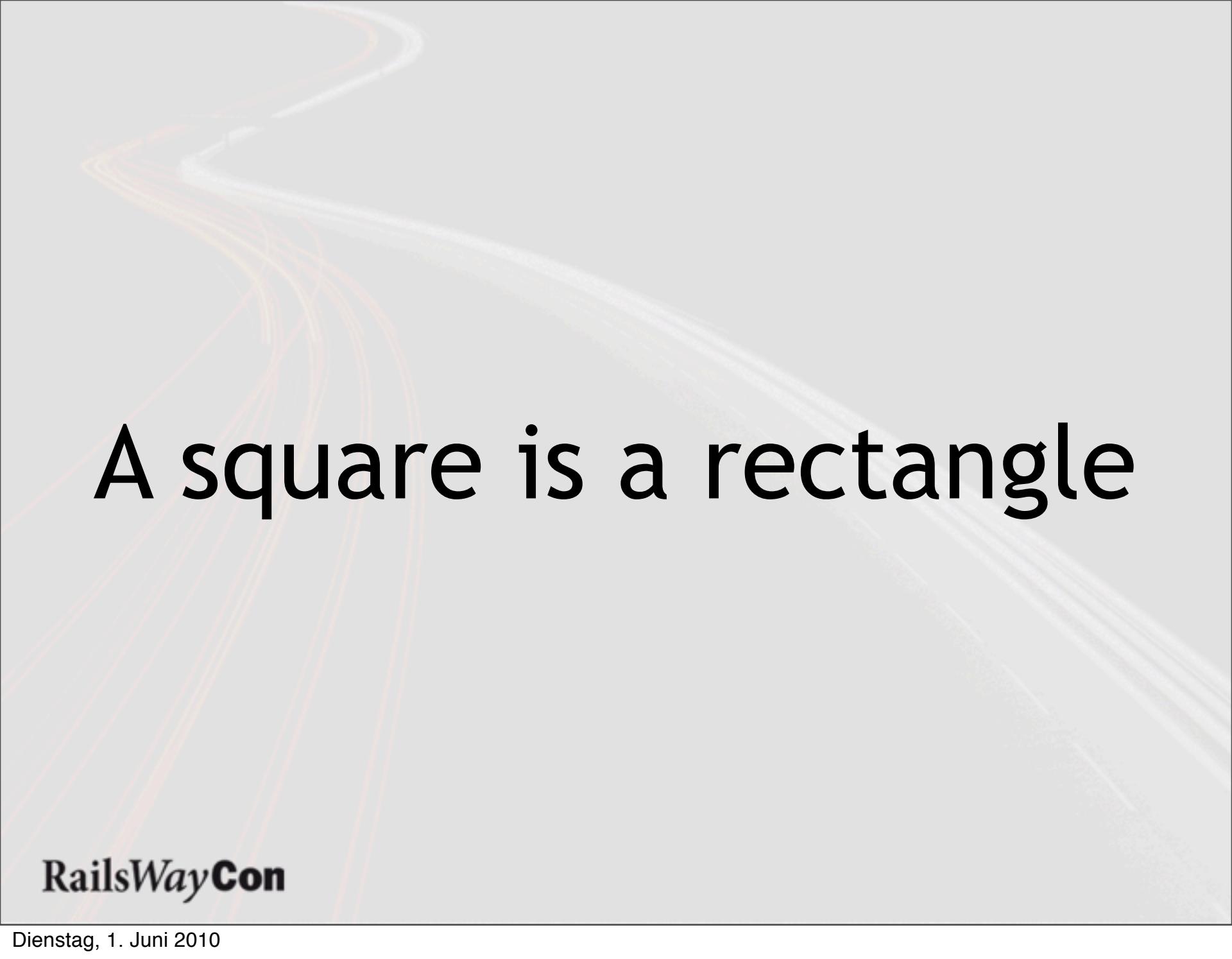
RailsWay**Con**

Wrong !

RailsWay**Con**

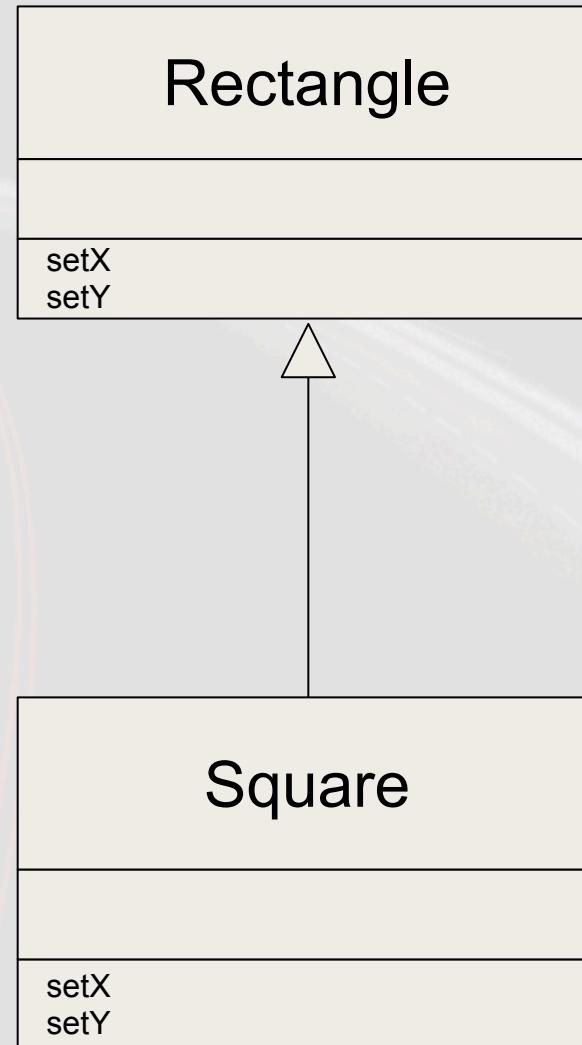
The classic violation

RailsWay**Con**



A square is a rectangle

RailsWayCon



Rectangle

```
>> class Rectangle
>>   attr_accessor :width, :height
>> end
=> nil
>>
?> shape = Rectangle.new
=> #<Rectangle:0x10114fad0>
>> shape.width
=> nil
>> shape.width=3
>> shape.width
=> 3
>> shape.height=5
>> shape.height
=> 5
>> shape.width
=> 3
```

Square

```
>> class Square
?>   def width
>>     @dimension
>>   end
?>   def height
>>     @dimension
>>   end
?>   def width= n
>>     @dimension = n
>>   end
?>   def height= n
>>     @dimension = n
>>   end
>> end

?> shape = Square.new
=> #<Square:0x101107e88>
?> puts shape.width
nil
?> shape.width=3
=> 3
?> shape.width
=> 3
?> shape.height
=> 3
```

A Problem...

```
>> s = [Rectangle.new, Square.new]
=> [#<Rectangle:0x1005642e8>, #<Square:0x100564298>]
>> a_rectangle = s[rand(2)]
=> #<Square:0x100564298>
>> a_rectangle.height=1
=> 1
>> a_rectangle.width=3
=> 3
>> a_rectangle.height
=> 3
```

CCD Common Conceptual Denominator

RailsWay**Con**

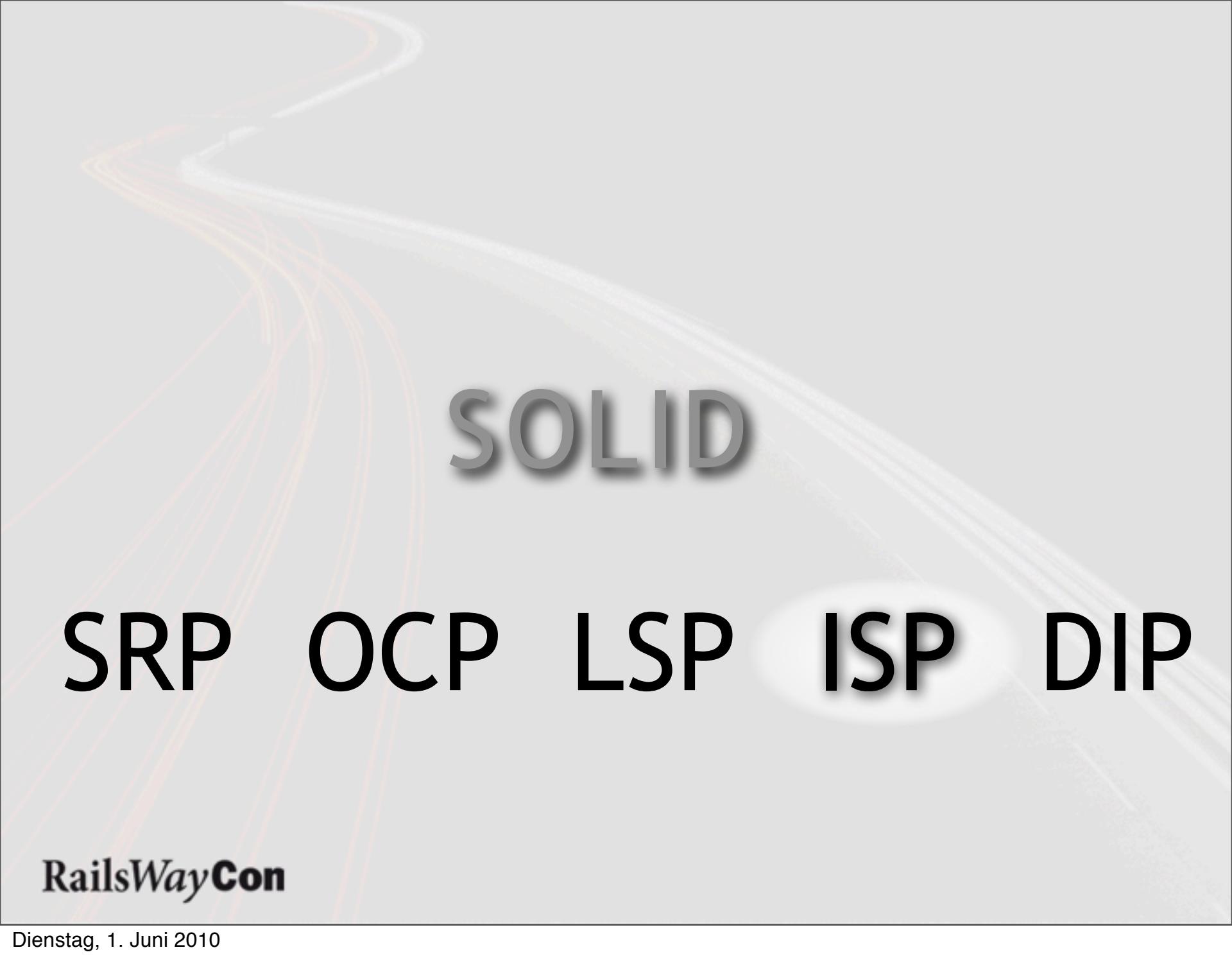


dup

RailsWay**Con**

```
irb 1:0> 5.respond_to? :dup
=> true
irb 2:0> 5.dup
TypeError: can't dup Fixnum
          from (irb):1:in `dup'
          from (irb):1
irb 3:0>
```

[http://blog.objectmentor.com/articles/2007/02/17/
liskov-substitution-principle-and-the-ruby-core-libraries](http://blog.objectmentor.com/articles/2007/02/17/liskov-substitution-principle-and-the-ruby-core-libraries)



SOLID

SRP OCP LSP ISP DIP

RailsWay**Con**

ISP

Interface Segregation Principle



Make fine grained
interfaces that are
client specific.

RailsWayCon



RailsWay**Con**

Dienstag, 1. Juni 2010

Users Controller

```
class UsersController < ApplicationController

  ssl_required :new, :create, :edit, :update, :destroy, :activate,
  :change_password, :forgot_password, :reset_password, :make_profile,
  :my_contacts
  ssl_allowed :eula, :index, :show

  access_control
  [:suspend, :unsuspend, :destroy, :purge, :delete, :admin, :ban, :remove_ban] =>
  'admin'

  before_filter :find_user

  skip_after_filter :store_location

  def show
    unless @user == current_user
      redirect_to access_denied_path(@locale)
    else
      respond_to do |format|
        format.html
        format.js { render :partial => "users/#{@context.title}/#{@partial}" }
      end
    end
  end
end

...
```

more UsersController

```
def activate
  logout_keeping_session!
  user = User.find_by_activation_code(params[:activation_code]) unless
    params[:activation_code].blank?

  case
  when (!params[:activation_code].blank?) && user && !user.active?
    user.activate!
    flash[:notice] = t(:message_sign_up_complete)
    unless params[:context].blank?
      redirect_to login_path(:context => params[:context])
    else
      redirect_to "/login"
    end
  when params[:activation_code].blank?
    flash[:error] = t(:message_activation_code_missing)
    redirect_back_or_default("/")
  else
    flash[:error] = t(:message_user_with_that_activation_code_missing)
    redirect_back_or_default("/")
  end
end
```

User Class Revisited

```
class User < ActiveRecord::Base
  ...
end

class Registration < ActiveRecord::Base
  set_table_name "users"

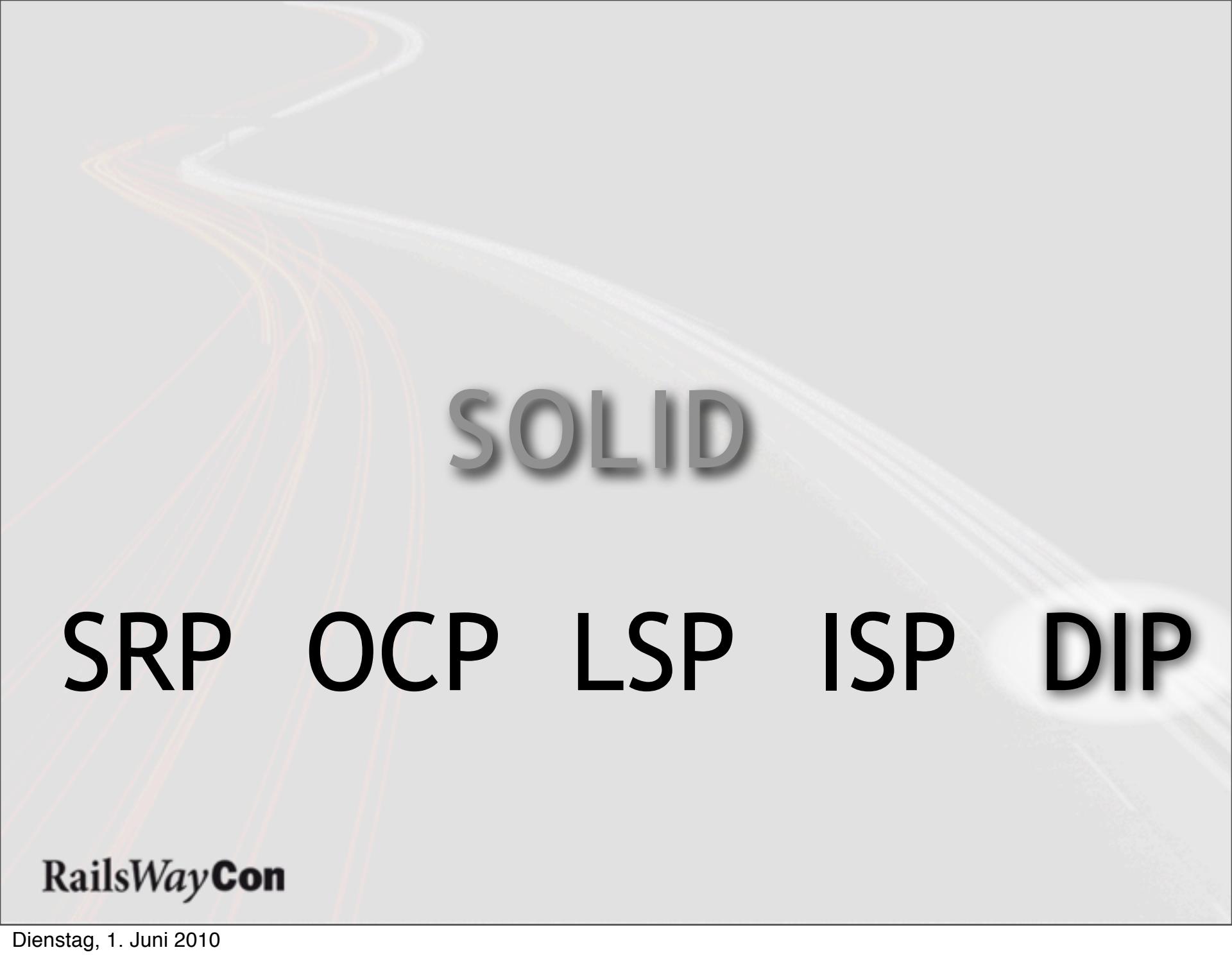
  acts_as_state_machine :initial => :pending

  state :pending, :enter => :make_activation_code
  state :active,  :enter => :do_activate
  ...

  event :activate do
    transitions :from => :pending, :to => :active
  end
  ...
end
```

```
class RegistrationController < ApplicationController
...
def activate
  logout_keeping_session!
  code_is_blank = params[:activation_code].blank?
  registration = Registration.find_by_activation_code(params[:activation_code]) unless code_is_blank

  case
  when (!code_is_blank) && registration && !registration.active?
    registration.activate!
    flash[:notice] = t(:message_sign_up_complete)
    unless params[:context].blank?
      redirect_to login_path(:context => params[:context])
    else
      redirect_to "/login"
    end
  when code_is_blank
    flash[:error] = t(:message_activation_code_missing)
    redirect_back_or_default("/")
  else
    flash[:error] = t(:message_user_with_that_activation_code_missing)
    redirect_back_or_default("/")
  end
end
...
end
```



SOLID

SRP OCP LSP ISP DIP

RailsWayCon

DIP

Dependency Inversion Principle



Depend on
abstractions, not on
concretions.

RailsWay**Con**



RailsWay**Con**

From our OCP example to DIP

```
out = Outputer.new  
out.puts "Testing"
```

The code we wish we had

```
class TronBot
  def initialize
    @@out = TRON_ENVIRONMENT[:debugger]
  end

  def some_method
    ...
    @@out.puts "Testing"
    ...
  end

end
```

TSTTCPW

```
TRON_ENVIRONMENT = {
  :debugger => Outputer.new ($stderr),
  :game_engine => Outputer.new ($stdout),
  :user_io => Outputer.new ($stderr)
}
```

Later...

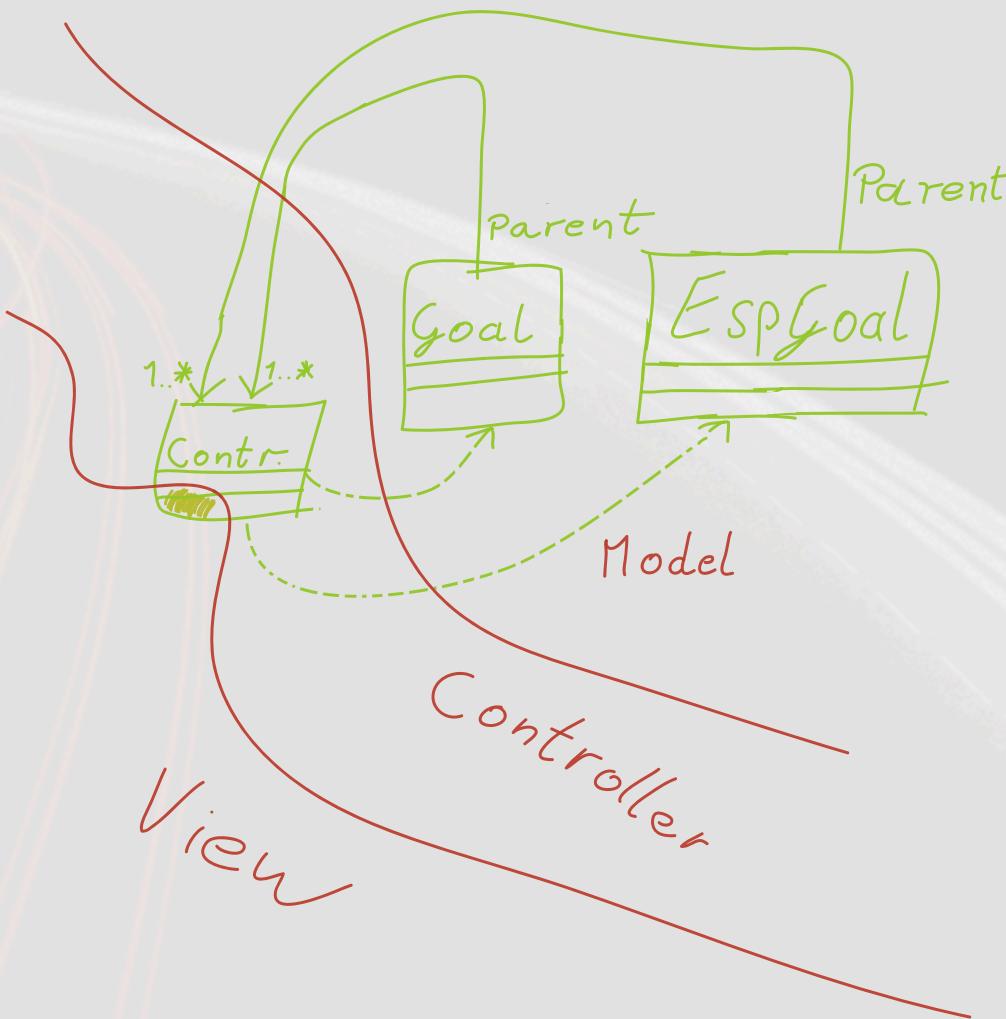
```
TRON_ENVIRONMENT = {
  :debugger => Outputer.new ($stderr),
  :game_engine => Outputer.new (TCP_OUTPUTTER),
  :user_io => Outputer.new ($stderr)
}
```

DIP Violation in Controller

```
format.js do
  render :update do |page|
    if @parent_object.class == EspGoal
      @esp_goal_descriptor = @current_object
      page.replace_html "descriptor_#{@current_object.id}",
        :partial => "edit_esp_goal_descriptor",
        :locals => { :esp_goal_descriptor => @esp_goal_descriptor,
                      :parent_object => @parent_object}
    else
      @goal_descriptor = @current_object
      page.replace_html "descriptor_#{@current_object.id}",
        :partial => "edit_goal_descriptor",
        :locals => { :goal_descriptor => @goal_descriptor,
                      :parent_object => @parent_object}
    end
  end
end
```

DIP Violation in Controller

```
format.js do
  render :update do |page|
    if @parent_object.class == EspGoal
      @esp_goal_descriptor = @current_object
      page.replace_html "descriptor_#{@current_object.id}",
        :partial => "edit_esp_goal_descriptor",
        :locals => { :esp_goal_descriptor => @esp_goal_descriptor,
                     :parent_object => @parent_object}
    else if @parent_object.class == Goal
      @goal_descriptor = @current_object
      page.replace_html "descriptor_#{@current_object.id}",
        :partial => "edit_goal_descriptor",
        :locals => { :goal_descriptor => @goal_descriptor,
                     :parent_object => @parent_object}
    else if @parent_object.class == LearningGoal
      ...
      ...
    end
  end
end
```

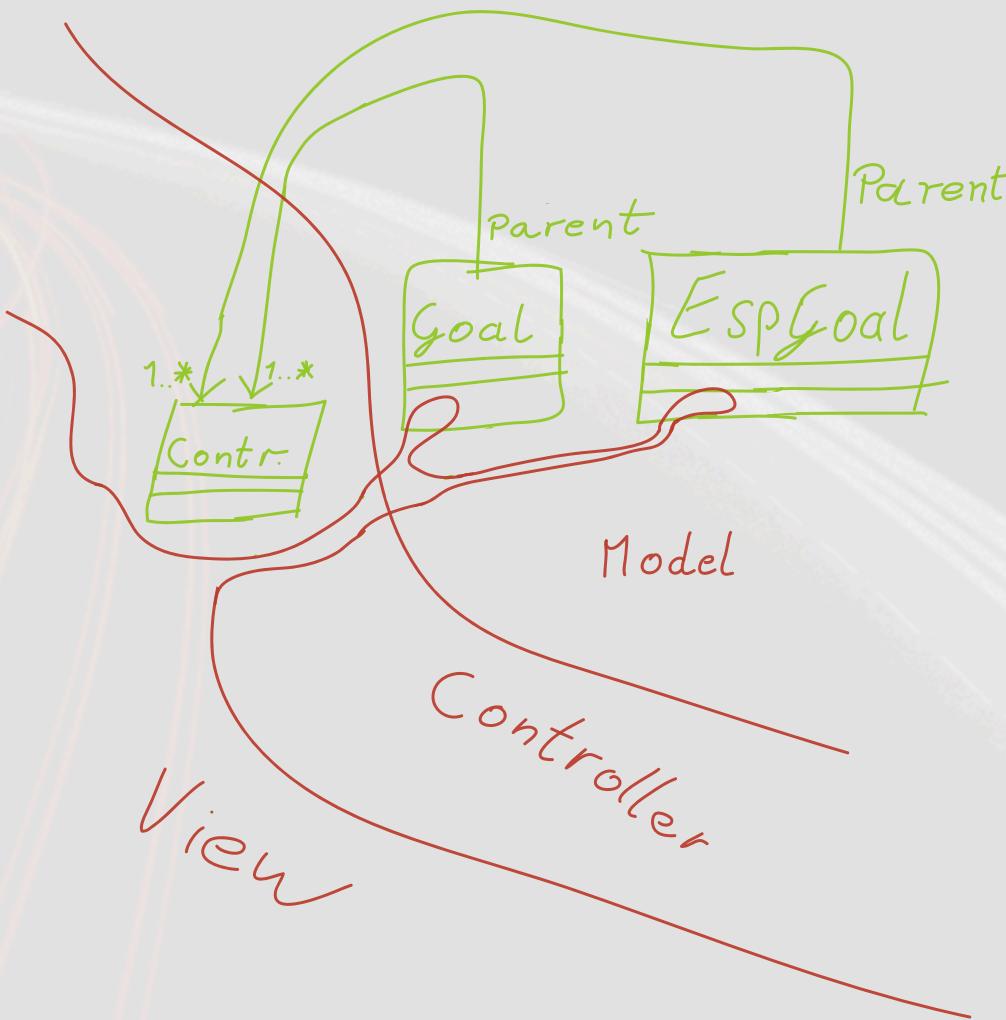


1st Refactoring

```
def show
...
format.js do
  render :update do |page|
    page.replace_html "descriptor_#{@current_object.id}",
                      @parent_object.page_replacement(@current_object)
  end
end
end

class EspGoal
  def page_replacement child
    { :partial => "edit_esp_goal_descriptor",
      :locals => { :esp_goal_descriptor => child,
                   :parent_object => self} }
  end
end

class Goal
  def page_replacement child
    { :partial => "edit_goal_descriptor",
      :locals => { :goal_descriptor => child,
                   :parent_object => self} }
  end
end
```



2nd Refactoring (Behaviour)

```
class EspGoalReplacmenent
  def self.my_class_sym
    EspGoal.to_sym
  end
  def partial_definition child
    { :partial => "edit_esp_goal_descriptor",
      :locals => { :esp_goal_descriptor => child,
                   :parent_object => child.esp_goal} }
  }
end
end

class GoalReplacmenent
  def self.my_class_sym
    Goal.to_sym
  end
  def partial_definition child
    { :partial => "edit_goal_descriptor",
      :locals => { :goal_descriptor => child,
                   :parent_object => child.goal} }
  }
end
end
```

2nd Refactoring (wiring)

```
class PartialContainer
  def add class_symbol, partial_replacement
    @@partial_replacements.add( class_symbol => partial_replacement)
  end

  def self.partial_replacement an_object
    unless @@partial_replacements
      self.add( EspGoalReplacement.my_class_sym, EspGoalReplacement.new)
      self.add( GoalReplacement.my_class_sym, GoalReplacement.new)
    end
    @@partial_replacement[an_object.class]
  end
end
```

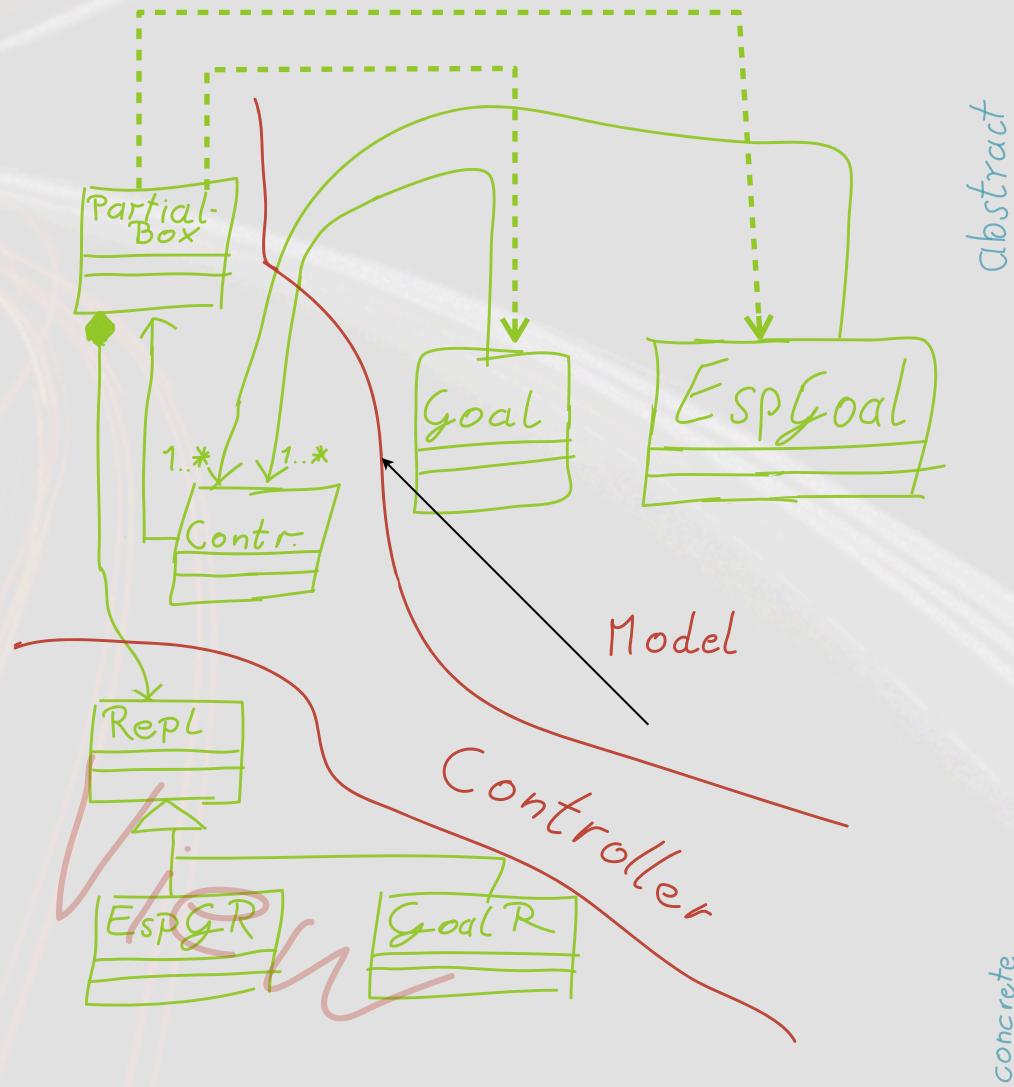
DIP Violation in Controller

```
format.js do
  render :update do |page|
    if @parent_object.class == EspGoal
      @esp_goal_descriptor = @current_object
      page.replace_html "descriptor_#{@current_object.id}",
:partial => "edit_esp_goal_descriptor",
      :locals => {:esp_goal_descriptor => @esp_goal_descriptor,
                  :parent_object => @parent_object}
    else
      @goal_descriptor = @current_object
      page.replace_html "descriptor_#{@current_object.id}",
:partial => "edit_goal_descriptor",
      :locals => {:goal_descriptor => @goal_descriptor,
                  :parent_object => @parent_object}
    end
  end
end
```

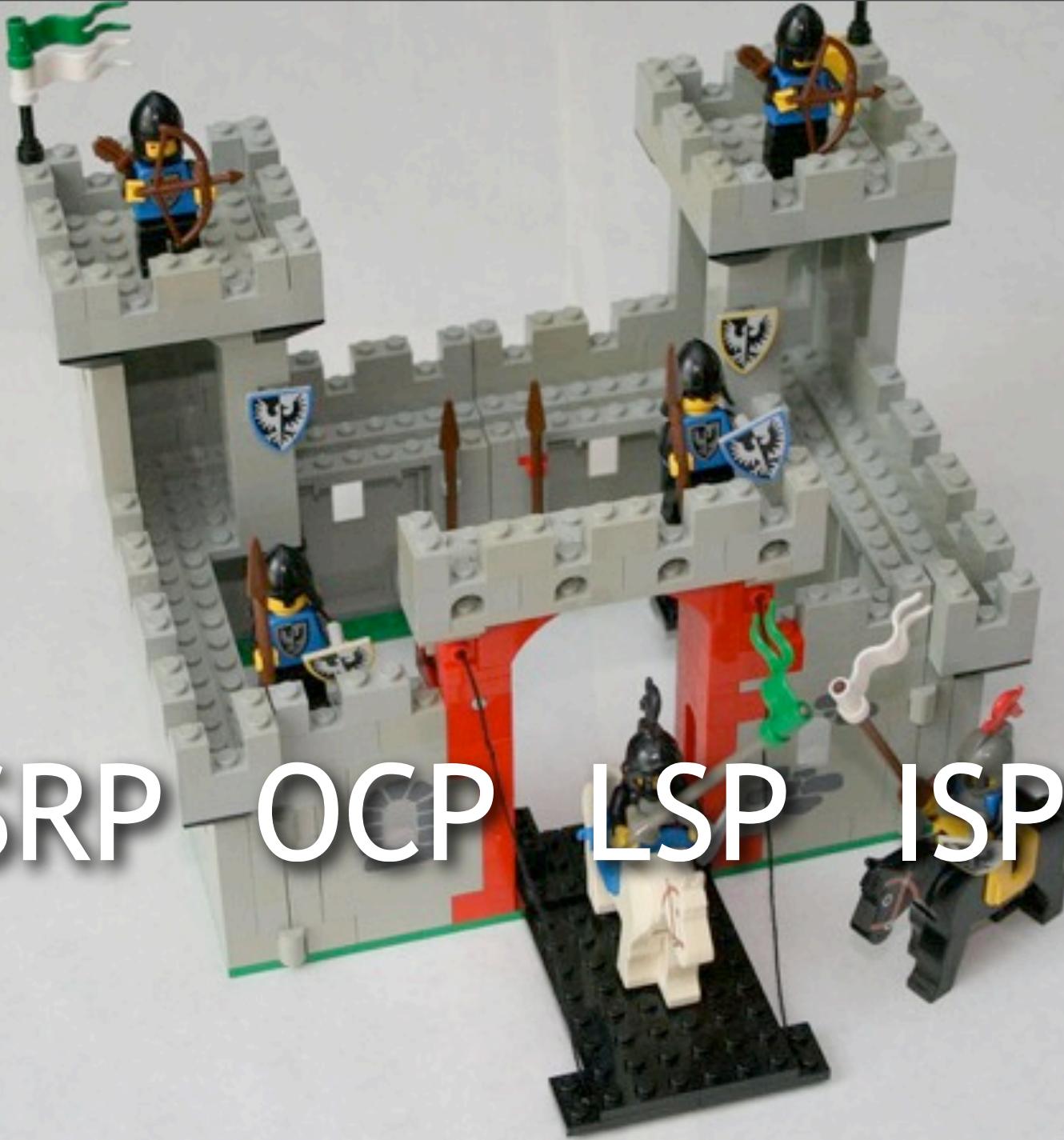
2nd Refactoring

- the Controller -

```
def show
  ...
  format.js do
    render :update do |page|
      page.replace_html "descriptor_#{@current_object.id}",
        PartialContainer.partial_replacement(@parent_object).
          partial_definition(@current_object)
    end
  end
end
```



SOLID
SRP OCP LSP ISP DIP



SRP OCP LSP ISP DIP



Questions?

SOLID



SRP OCP LSP ISP DIP

Domo Arigato

RailsWay**Con**

Coding Dojo

Wednesday 11:10

Salon 3

The Way of the carpenter is to become proficient in the use of his tools, first to lay his plans with a true measure and then perform his work according to plan.

– Go Rin No Sho

Credits (1/2)

PPP-Article (online)

<http://butunclebob.com/ArticleS.UncleBob.PrinciplesOfOOD>

Photos

<http://www.flickr.com/photos/dieterkarner/370967891/>

<http://www.flickr.com/photos/popcorncx/2221630487/sizes/l/>

<http://www.flickr.com/photos/bdesham/2432400623/>

<http://www.flickr.com/photos/popcorncx/2221630487/>

<http://www.flickr.com/photos/glennbatuyong/4081599002/in/photostream/>

<http://www.flickr.com/photos/glennbatuyong/4081599168/in/photostream/>

<http://www.flickr.com/photos/renfield/3865907619/>

Credits (2/2)

Photos

<http://www.flickr.com/photos/renfield/3865907619/>

<http://www.flickr.com/photos/maxpower/5160699/>

http://programmer.97things.oreilly.com/wiki/index.php/Uncle_Bob

<http://www.flickr.com/photos/georgivar/3288942086/>

<http://www.everystockphoto.com/photo.php?imageId=237523>

<http://www.flickr.com/photos/pasukaru76/3992935923/>

Lizenz

[http://creativecommons.org/licenses/by-sa/
3.0/de/](http://creativecommons.org/licenses/by-sa/3.0/de/)



Attribution-ShareAlike 3.0 Germany

Jens-Christian Fischer
InVisible GmbH

@jcfischer
jens-christian@invisible.ch
http://blog.invisible.ch

Michael Mahlberg
Consulting Guild AG

@MMahlberg
mm@michaelmahlberg.de
http://agile-aspects.blogspot.com

Get the Presentation

<http://>

