

Web-Oriented Software Development Ruby on Rails (RoR)



André Santos

firstname.lastname@tut.fi

Outline

1. Motivation
2. Basic Idea
3. Highlights
4. Demo
5. Conclusions

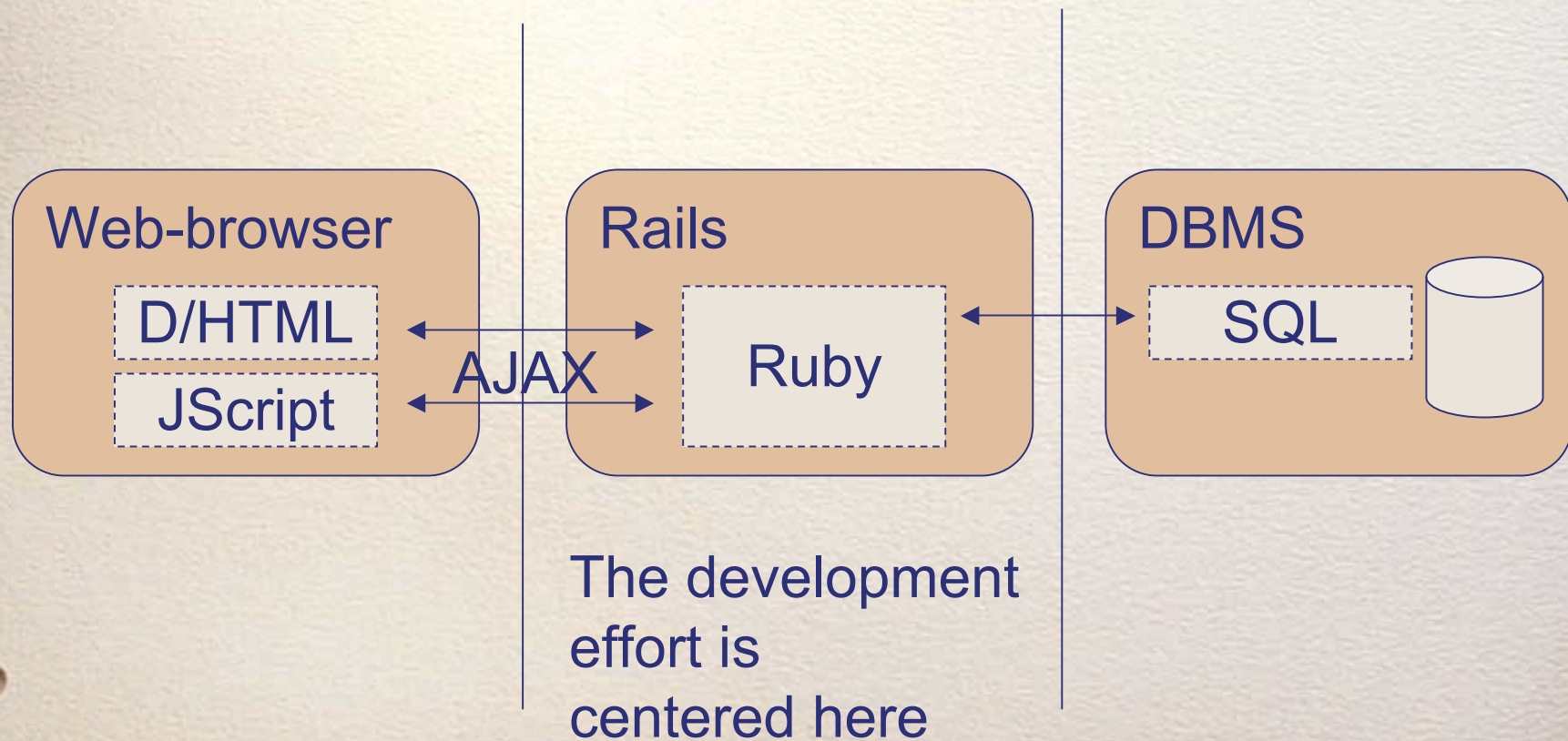
1. Motivation

- Complex configuration files
- Multiplicity of languages (e.g. HTML, JAVASCRIPT, XML, PHP, SQL)
- Non-effective domain-centered development
- Repetitive and error-prone development

2. Basic Idea

- *Ruby on Rails (RoR)*
 - “Ruby” is an object-oriented language
 - “Rails” is a platform/framework which imposes a MVC (Model View Controller) architecture for database-backed web applications
 - Open source
- The development is done using Ruby, Rails scripts, and following the platform’s rules (structure, conventions, etc)
- Principle DRY - *Don’t Repeat Yourself*

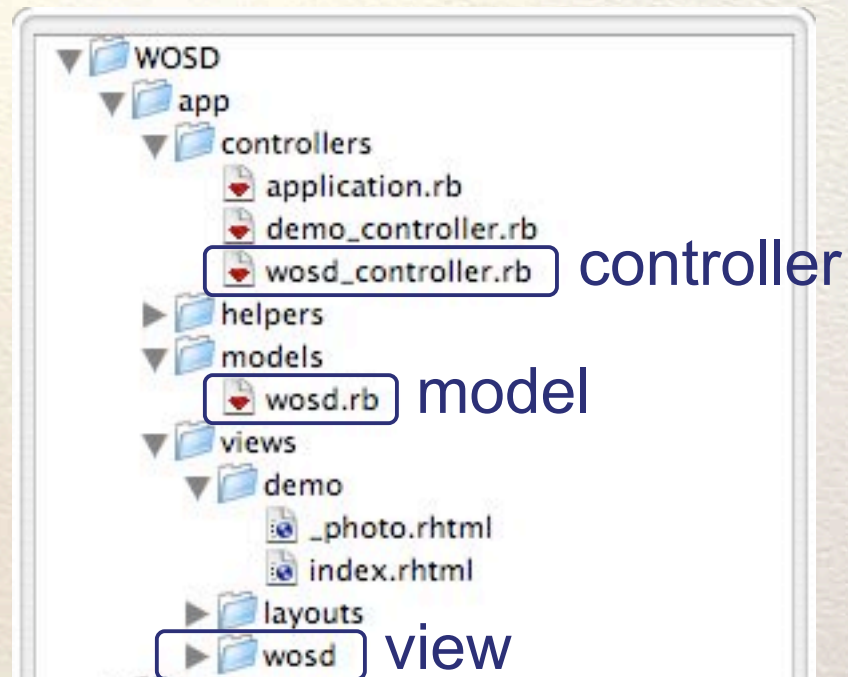
3-tier MVC architecture



3. Highlights

- Impose an MVC architecture in an easy fashion (e.g. comparing to J2EE)
 - *structure, generators, conventions*
- High reduction of lines of code
 - *high-level programming primitives, conventions*
- Simplify configuration issues
 - *naming conventions*
- Domain-centered development
 - *by having the above...*

Application structure - MVC



...

Reduction of lines of code

model

```
create table wosds (  
  id INTEGER PRIMARY KEY,  
  name VARCHAR(100));
```



```
class Wosd < ActiveRecord::Base  
end
```

controller

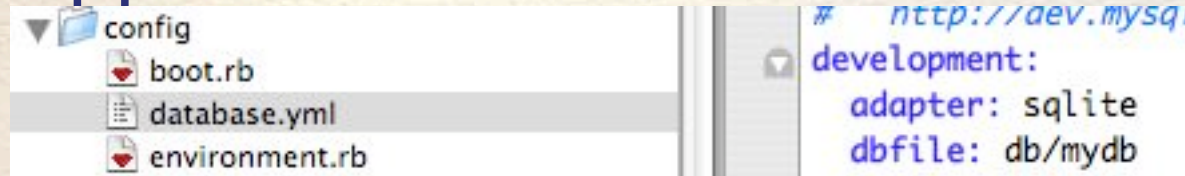
```
class WosdController < ApplicationController  
  scaffold :wosd  
end
```

view

```
<P><%= text_field 'wosd', 'name' %></P>
```


Configuration

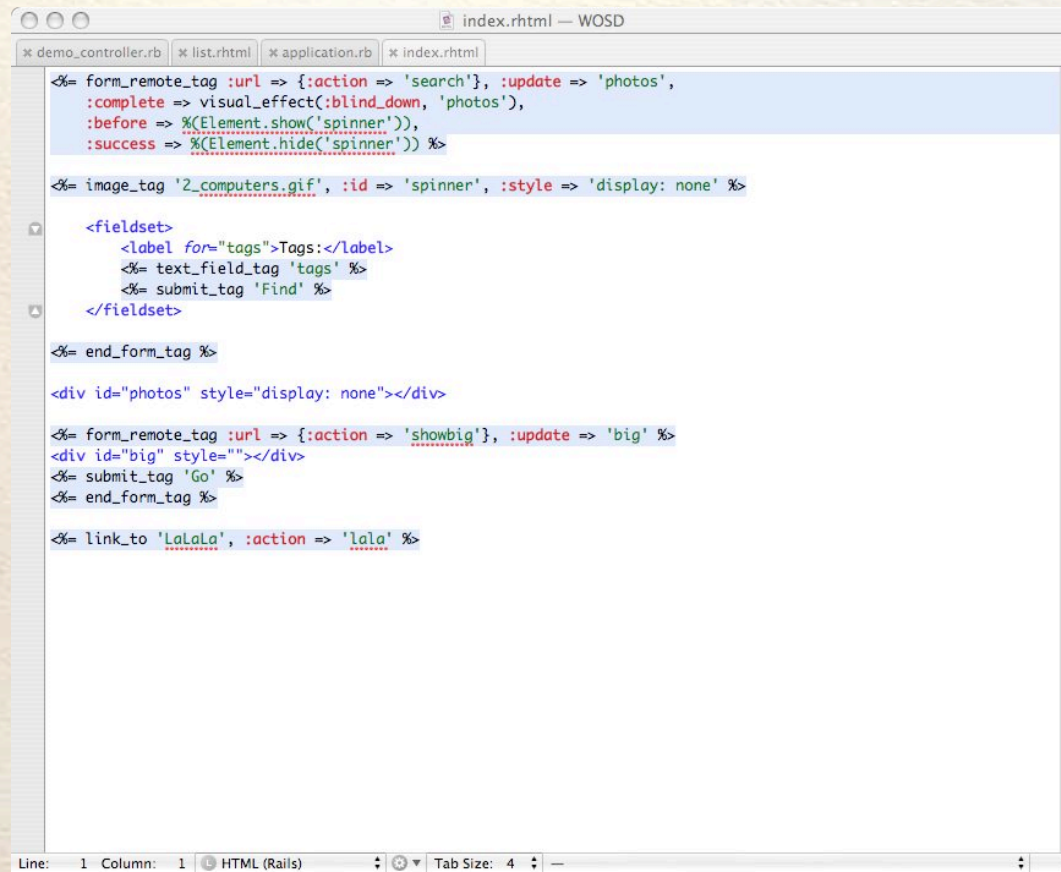
- RoR imposes the directory structure for the application files and their names



- Example of convention: the class name *Wosd* corresponds to a database table with that name

```
class Wosd < ActiveRecord::Base
end
```


Demo



The screenshot shows a web browser window titled "index.rhtml — WOSD". The browser has several tabs open: "demo_controller.rb", "list.rhtml", "application.rb", and "index.rhtml". The main content area displays a form with a search button and a "showbig" button. The form is rendered using Rails' form helpers, including `form_remote_tag`, `image_tag`, `fieldset`, `label`, `text_field_tag`, `submit_tag`, `end_form_tag`, `div`, `link_to`, and `end_form_tag`. The form has a search button with a spinner and a "showbig" button. The "showbig" button is a simple submit button labeled "Go". The form is styled with a "display: none" attribute. The browser's status bar at the bottom shows "Line: 1 Column: 1 HTML (Rails) Tab Size: 4".

```
<%= form_remote_tag :url => {:action => 'search'}, :update => 'photos',  
:complete => visual_effect(:blind_down, 'photos'),  
:before => %(Element.show('spinner')),  
:success => %(Element.hide('spinner')) %>  
  
<%= image_tag '2_computers.gif', :id => 'spinner', :style => 'display: none' %>  
  
<fieldset>  
  <label for="tags">Tags:</label>  
  <%= text_field_tag 'tags' %>  
  <%= submit_tag 'Find' %>  
</fieldset>  
  
<%= end_form_tag %>  
  
<div id="photos" style="display: none"></div>  
  
<%= form_remote_tag :url => {:action => 'showbig'}, :update => 'big' %>  
<div id="big" style=""></div>  
<%= submit_tag 'Go' %>  
<%= end_form_tag %>  
  
<%= link_to 'LaLaLa', :action => 'lala' %>
```


Conclusions

- Like any framework, the productivity gains are only noticed after its learning
 - however, with RoR at least the basics are achieved pretty fast
- Like in any framework, the flexibility is limited to the imposed architecture
 - however, if “all” web applications should be based on MVC... Why not?
- Some of the imposed naming conventions seem not to be very obvious/natural
 - E.g. SQL Table *wosds* corresponds to Ruby Model *Wosd*
- Ruby is an interpreted language (not so efficient)
 - however, it may significantly improve development productivity

Examples

- Online systems which are built on RoR:
 - <http://www.basecamphq.com/>
 - <http://www.strongspace.com/>
 - <http://www.campfirenow.com/>

References

- Ruby on Rails website:
<http://www.rubyonrails.org/>
- *Agile Web Development with Rails:*
<http://www.pragmaticprogrammer.com/titles/rails/index.html>

Questions?
