Hey, it's HAML!

Markup should be beautiful.

The Plan

- Introduction & Basics

- Converting ERB to HAML

- Summary & Resources

- Q & A

Introduction & Basics

What is HAML?

- Templating engine

- Easier than HTML (ERB)

- Eliminates redundancy

- Reflects underlying structure

Core principles

- Markup should be beautiful

- Markup should be DRY

- Markup should be well-indented

- XHTML structure should be clear

Using HAML

```
$ gem install merb-more

use_template_engine :haml          config/init.rb

dependency 'merb-haml'             config/dependencies.rb

template_name.html.haml            app/views/...
```

Basics

```
%tagname{:attr => 'value'} Contents

%tagname#id.class
```

```
#foo Hello
```

```
<div id='foo'>Hello</div>
```

```
.bar World!
```

```
<div class='bar'>World!</div>
```

Indentation

```
%ul
  %li Salt
  %li Pepper
```

```
<ul>
  <li>Salt</li>
  <li>Pepper</li>
</ul>
```

More indentation

```
%p
   Hello,
   World!
```

`<p>`Hello, World!`</p>`

```
Haml::Template::options[:ugly] = true          config/init.rb
```

Embedded Ruby

```
%p
  Date/Time:
  - now = DateTime.now
  %b
    - if now > DateTime.parse('December 31, 2008')
      = "Happy New Year!"
    - else
      = now.to_s
```

```
<p>Date/Time:<b>2008-11-27T19:06:28-05:00<b></p>
```

```
<p>Date/Time:<b>Happy New Year!</b></p>
```

The Finer Points

%, {} and /

The forward slash character, when placed at the end of a tag definition, causes the tag to be self-closed.

```
%bar/
```

```
<bar/>
```

```
%head{:name => "doc_head"}
  %script{:type => "text/" + "javascript", |
          :src => "javascripts/script_#{2 + 7}"} |
```

```
<head name='doc_head'>
  <script src='/docs/rdoc/javascripts/script_9' type='text/javascript'>
  </script>
</head>
```

{} contents

A Ruby method call that returns a hash can be substituted for the hash contents.

```ruby
def hash1
  {:bread => 'white', :filling => 'peanut butter and jelly'}
end

def hash2
  {:bread => 'whole wheat'}
end
```

```haml
%sandwich{hash1, hash2, :delicious => true}/
```

```html
<sandwich bread='whole wheat' delicious='true' filling='peanut butter and jelly' />
```

[]

```ruby
# file: app/controllers/users.rb
def show
  @user = User.get(15)
end

-# file: app/views/users/show.html.haml
%div[@user, :greeting]
  %bar[20]/
  Hello!
```

```html
<div class='greeting_user' id='greeting_user_15'>
    <bar class='fixnum' id='fixnum_20' />
    Hello!
</div>
```

=, ==, and ~

```
%p= "Hello"
```

```
<p>Hello</p>
```

```
%p== Hello, #{"Merb"}!
```

```
<p>Hello, Merb!</p>
```

```
~ "Foo\n<pre>Bar\nBaz</pre>"
```

```
Foo
<pre>Bar&#x000A;Baz</pre>
```

!!!

When describing XHTML documents with Haml, you can have a document type or XML prolog generated automatically by including the characters !!!.

```
!!! Strict
```

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

```
!!! XML iso-8859-1
```

```
<?xml version='1.0' encoding='iso-8859-1' ?>
```

/ revisited

```
%peanutbutterjelly
    / This is the peanutbutterjelly element
    I like sandwiches!
```

```
<peanutbutterjelly>
  <!-- This is the peanutbutterjelly element -->
  I like sandwiches!
</peanutbutterjelly>
```

```
/
  %p This doesn't render...
    %div
      %h1 Because it's commented out!
```

```
<!--
  <p>This doesn't render...</p>
  <div>
    <h1>Because it's commented out!</h1>
  </div>
-->
```

Summary & Resources

Resources

🚀 http://github.com/schulty/merbday   slides and code
http://bit.ly/haml   scribd slideshow

Q & A