

Radiant Extensions

Marty Haught
Haught Consulting

What are extensions?

- Vertical slices of a Rails application
- Customize nearly any aspect of Radiant

Why extensions?

- Minimal Radiant core
- Need a way to add functionality easily

How to install

- copy extension files into:
vendor/extensions/<extension_name>
 - svn:externals
 - svn export
 - unzip/tar

Creating an extension

```
s-MBP:~/demo marty$ script/generate extension PageEvent
create vendor/extensions/page_event/app/controllers
create vendor/extensions/page_event/app/helpers
create vendor/extensions/page_event/app/models
create vendor/extensions/page_event/app/views
create vendor/extensions/page_event/db/migrate
create vendor/extensions/page_event/lib/tasks
create vendor/extensions/page_event/test/fixtures
create vendor/extensions/page_event/test/functional
create vendor/extensions/page_event/test/unit
create vendor/extensions/page_event/README
create vendor/extensions/page_event/Rakefile
create vendor/extensions/page_event/page_event_extension.rb
create vendor/extensions/page_event/lib/tasks/page_event_ext
```

Extension layout

```
app
  controllers
  helpers
  models
  views
db
  migrate
lib
  tasks
test
  fixtures
  functionals
  unit
```

Initialization

page_event_extension.rb

```
class PageEventExtension < Radiant::Extension
  version "0.1"
  description "Allows you to add event dates to your pages that can be viewed on a site-wide calendar"
  url "http://martyhaught.com"

  define_routes do |map|
    map.connect 'admin/page_events/:action', :controller => 'page_events'
  end

  def activate
    Page.send :include, PageEventTags
    Page.send :include, PageEvent::PageExtensions
    admin.page.edit.add :layout_row, "edit_page_event"
    admin.tabs.add "Page Events", "/admin/page_events", :after => "Pages", :visibility => [:all]
  end

  def deactivate
    admin.tabs.remove "Page Events"
  end
end
```

Generating Models

```
marty$ script/generate extension_model PageEvent NewModelName title:string url:string description:text
```

- Generates:
 - model class
 - unittest
 - fixture
 - migration

Migrations

```
class CreatePageEvents < ActiveRecord::Migration
  def self.up
    add_column :pages, :event_datetime, :datetime
  end

  def self.down
    remove_column :pages, :event_datetime
  end
end
```

```
marty$ rake db:migrate:extensions
```

```
marty$ rake radiant:extensions:page_event:migrate VERSION=0
```

Extending Models

```
module PageEvent::PageExtensions

  def self.included(base)
    base.extend ClassMethods
  end

  module ClassMethods
    def events_by_month(date = Time.now)
      month_start = date.at_beginning_of_month
      Page.find(:all, :conditions => "event_datetime #{(month_start .. month_start.next_month - 1.minute).to_s(:db)}")
    end
  end
end
```

```
def activate
  Page.send :include, PageEventTags
  Page.send :include, PageEvent::PageExtensions
  admin.page.edit.add :layout_row, "edit_page_event"
  admin.tabs.add "Page Events", "/admin/page_events", :after => "Pages", :visibility => [:all]
end

def deactivate
```

Tests

```
require File.dirname(__FILE__) + '/../test_helper'

class PageEventExtensionTest < Test::Unit::TestCase
  fixtures :pages

  def test_initialization
    assert_equal File.join(File.expand_path(RAILS_ROOT), 'vendor', 'extensions', 'page_event'), PageEventExtension.root
    assert_equal 'Page Event', PageEventExtension.extension_name
  end

  def test_events_by_month
    assert_respond_to Page, :events_by_month
    assert_equal(3, Page.events_by_month.size)
    assert_not_equal(Page.find(:all).size, Page.events_by_month.size)
  end
end
```

Generating Controllers

```
marty$ script/generate extension_controller PageEvent admin/page_events
```

- Generates:
 - controller
 - functional test
 - views directory
 - controller helper

Activating Controllers

```
define_routes do |map|
  map.connect 'admin/page_events/:action', :controller => 'page_events'
end

def activate
  Page.send :include, PageEventTags
  Page.send :include, PageEvent::PageExtensions
  admin.page.edit.add :layout_row, "edit_page_event"
  admin.tabs.add "Page Events", "/admin/page_events", :after => "Pages", :visibility => [:all]
end

def deactivate
  admin.tabs.remove "Page Events"
end
```

Radius Tags

```
module PageEventTags
  include Radiant::Taggable

  tag "event" do |tag|
    tag.expand if tag.locals.page.event_datetime
  end

  tag "event:date" do |tag|
    tag.locals.page.event_datetime.strftime("%m/%d/%Y") if tag.locals.page.event_datetime
  end

  tag "event:time" do |tag|
    tag.locals.page.event_datetime.strftime("%I:%M %p") if tag.locals.page.event_datetime
  end
end
```

```
<div id="wrapper">
  <h2><r:title /></h2>
  <r:event>
    Date: <r:date /> <br>
    Time: <r:time /> <br>
  </r:event>
  <p><r:content /></p>
</div>
```

Shards

- Creates insertion points, 'regions', in admin

```
def activate
  Page.send :include, PageEventTags
  Page.send :include, PageEvent::PageExtensions
  admin.page.edit.add :layout_row, "edit_page_event"
  admin.tabs.add "Page Events", "/admin/page_events", :after => "Pages", :visibility => [:all]
end
```

```
<%= calendar_date_select_includes('red') %>
<p><label for="page_date_event_id">Event Date</label>
  <%= calendar_date_select :page, :event_datetime, :time => :mixed %>
</p>
```

Tips

- Look over other extensions
- Read Radiant source/tests/specs
- Start dabbling

Thanks!

- Marty Haught
- mghaught@gmail.com
- <http://martyhaught.com>
- <http://radiantcms.org>