

MERB

Kas yra MERB?



MERB

Kas yra MERB?

MERB – tai mažas ir greitas

RUBY WEB FRAMEWORK



MERB

Kam jo reikia?



MERB



MERB sukurè

Ezra Zygmuntowicz



MERB

MERB yra
THREAD-SAFE



MERB

su MERB Jūs patys pasirenkate:

- Kokį ORM naudoti (DataMapper, Sequel, ActiveRecord, ...)
- Javascript framework(Prototypejs, JQuery, ...)
- Šablonų sistema(Erubis, HAML, ...)
- Testing framework(Rspec, Test::Unit, test/spec, ...)



MERB

merb-core

- HTTP abstraction
- Controllers
- Views
- Erubis(3x greitesnis už ERB)
- Routes
- REST



MERB merb-core

Rack Webserver Abstraction Layer

- Ebb
- Evented Mongrel
- FastCGI
- Mongrel
- Thin
- Webrick



MERB

```
config/rack.rb
```

```
run Merb::Rack::Application.new
```



MERB

config/rack.rb

```
rack_app = Proc.new do |env|  
  [200, {"Content-Type" => "text/html"},  
  "hello world!"]  
end  
run rack_app
```



MERB

```
uploader = Proc.new do |env|
  request = Merb::Request.new(env)
  if request.path =~ /^\/images\/upload\/
    params = request.params
    FileUtils.mv params[:file][:tempfile].path,
                 Merb.root / 'public' / 'uploads' / params[:file][:filename]
    headers = {"Content-Type"=>"text/html",
              "Location" => "/images"}
    [302, headers, "You are being redirected"]
  else
    [404, {}, "No Upload here, move along"]
  end
end
```

```
merb = Merb::Rack::Application.new
run Rack::Cascade.new([uploader, merb])
```



MERB

Visas papildomas funkcionalumas
sudētas i plugin'us



MERB

merb-more

- merb-action-args
- merb-assets
- merb-builder
- merb-cache
- merb-gen
- merb-haml
- merb-mailer
- merb-parts



MERB

merb-gen

merb-gen app app_name (for a standard application)
merb-gen app app_name --flat (for a flattened application)
merb-gen app app_name --very-flat (for a single file application)
merb-gen frozen-merb (script to run from frozen gems/framework)
merb-gen plugin plugin_name (creates a plugin gem skeleton)



MERB

merb app

app

controllers

helpers

views

models

config

gems

lib

log

public

spec



MERB

merb-gen generators

controller, freezer, model, part_controller,
resource, resource_controller



MERB

app/config

- environments
- init.rb
- rack.rb
- router.rb



MERB

config/init.rb

```
c[:session_secret_key] = '6cd6 ...'
```

```
c[:session_store] = 'cookie'
```

```
# use_test :test_unit
```

```
# use_test :rspec
```

```
### Uncomment for DataMapper ORM
```

```
# use_orm :datamapper
```

```
dependencies "RedCloth" => "> 3.0",
```

```
"ruby-aes-cext" => "= 1.0"
```

```
### Uncomment for ActiveRecord ORM
```

```
# use_orm :activerecord
```

```
### Uncomment for Sequel ORM
```

```
# use_orm :sequel
```



MERB

Routes



MERB

Path matching

```
r.match("/books/:book_id/:action").  
  to(:controller => "books")
```



MERB

Named Routes

```
r.match("/contact").
```

```
  to(:controller => "info", :action => "contact").
```

```
  name(:contact)
```

```
url(:contact) # => "/contact"
```



MERB

Namespace

```
r.namespace :admin do |admin|  
  admin.match("/roles").  
    to(:controller => "roles", :action => "index").  
    name(:roles)  
end  
  
url(:admin_roles) # => "/admin/roles"
```



MERB

Resources

```
r.resources :posts
```

```
url(:new_post) # => "/posts/new"
```



MERB Nested Resources

```
Merb::Router.prepare do |r|  
  r.resources :blogposts do |b|  
    b.resources :comments do |c|  
      c.resources :versions  
    end  
  end  
end  
end
```



MERB

Special Route

```
it "should support inbound user agents" do
  Merb::Router.prepare do |r|
    r.match(%r[^/foo/(.+)], :user_agent => /(MSIE|Gecko)/).
      to(:controller => "foo", :title => "[1]", :action => "show", :agent => ":user_agent[1]")
    end
  route_to("/foo/bar", :user_agent => /MSIE/).should have_route(
    :controller => "foo", :action => "show", :title => "bar", :agent => "MSIE"
  )
end
```



MERB Controllers

```
class Books < Application
  provides :json, :xml
```

```
  def index
```

```
    @books = Book.all
```

```
    display @books
```

```
  end
```

```
  def show(id)
```

```
    @post = Post[id]
```

```
    render
```

```
  end
```

```
end
```



MERB Parts

```
class Foo < Application
  def foo
    render
  end
end
```

```
app/views/foo/foo.erb
<%= part Bar => :baz %>
```

```
app/parts/views/bar/baz.erb
<%= params[:hello] %>
<%= @x %>
```

```
class Bar < Merb::PartController
  before :foo

  def foo
    @x = 1
  end

  def baz
    render
  end
end
```



MERB vs Rails

- Paketų pasirinkimas
- Jokios automagijos
- Mažas ir greitas
- Mažai dokumentacijos
- Nestabilus
- Mažai view helperių



MERB

MERB + Rails

MERB galima naudoti operacijoms, kuriuos labai ilgai užtrunka. Pavyzdžiui failų siuntimui.



MERB

Models



DataMapper

Ruby Object Relational Mapper

Greitas, Thread-safe ir pilnas ypatybių



DataMapper

Viena eilutė duomenų bazėje
turi atitikti vieną objektą



DataMapper

```
@parent = Tree.find(:first,  
  :conditions => ['name = ?', 'bob'])
```

```
@parent.children.each do |child|  
  puts @parent.object_id == child.parent.object_id  
end
```



DataMapper

- Nēra schema.rb
- Nēra migraciju
- Išsaugo tik pakeistus atributus

VALIO!



DataMapper

Model

```
class Post < DataMapper::Base
  property :title, :string
  property :body, :text
  property :created_at, :datetime
end
```



DataMapper

Auto migracija

```
rake dm:db:automigrate
```



DataMapper

Tinginiavimas gali būti virtualus

```
class Animal < DataMapper::Base
  property :name, :string
  property :notes, :text, :lazy => false
end
```



DataMapper

Eager Loading

```
zoos = Zoo.all
```

```
first = zoos.first
```

```
first.exhibits # Pakrauna visus exhibit objektus
```

```
visiems Zoo objektams zoo kintamajame
```



DataMapper

Primary key

```
class Post < DataMapper::Base
  property :slug, :string, :key => true
end
```



DataMapper

Paieška

```
zoo = Zoo[1]
```

```
zoo = Zoo['DFW']
```

```
zoo = Zoo.first(:name => 'Luke')
```

```
zoos = Zoo.all
```

```
zoos = Zoo.all(:open => true)
```



DataMapper

Paieška

```
Exhibition.all(:run_time.gt => 2, :run_time.lt => 5)
```

gt – Daugiau

lt – Mažiau

gte - Daugiau arba lygu

lte - Mažiau arba lygu

not - Neigimas

like

in



DataMapper

Associations

```
class Post < DataMapper::Base
  has_many :comments
  has_and_belongs_to_many :categories
  belongs_to :author
end
```



DataMapper

Associations

Has Many

Has One

Belongs To

Has and Belongs to Many

Self-Referential Has And Belongs To Many

Has Many Through



DataMapper

Callbacks

Materialize

Create

Save

Validation

Update

Destroy



DataMapper

Paranoja

property :deleted_at, :datetime

- Trynimo metu atnaujina deleted_at lauką.
- Paieška praleidžia deleted_at laukus.



Nuorodos

- MERB (<http://www.merbivore.com/>)
- Ezra Zygmuntowicz's blog (<http://brainspl.at/>)
- DataMapper (<http://datamapper.org/>)
- Rack (<http://rack.rubyforge.org/>)

