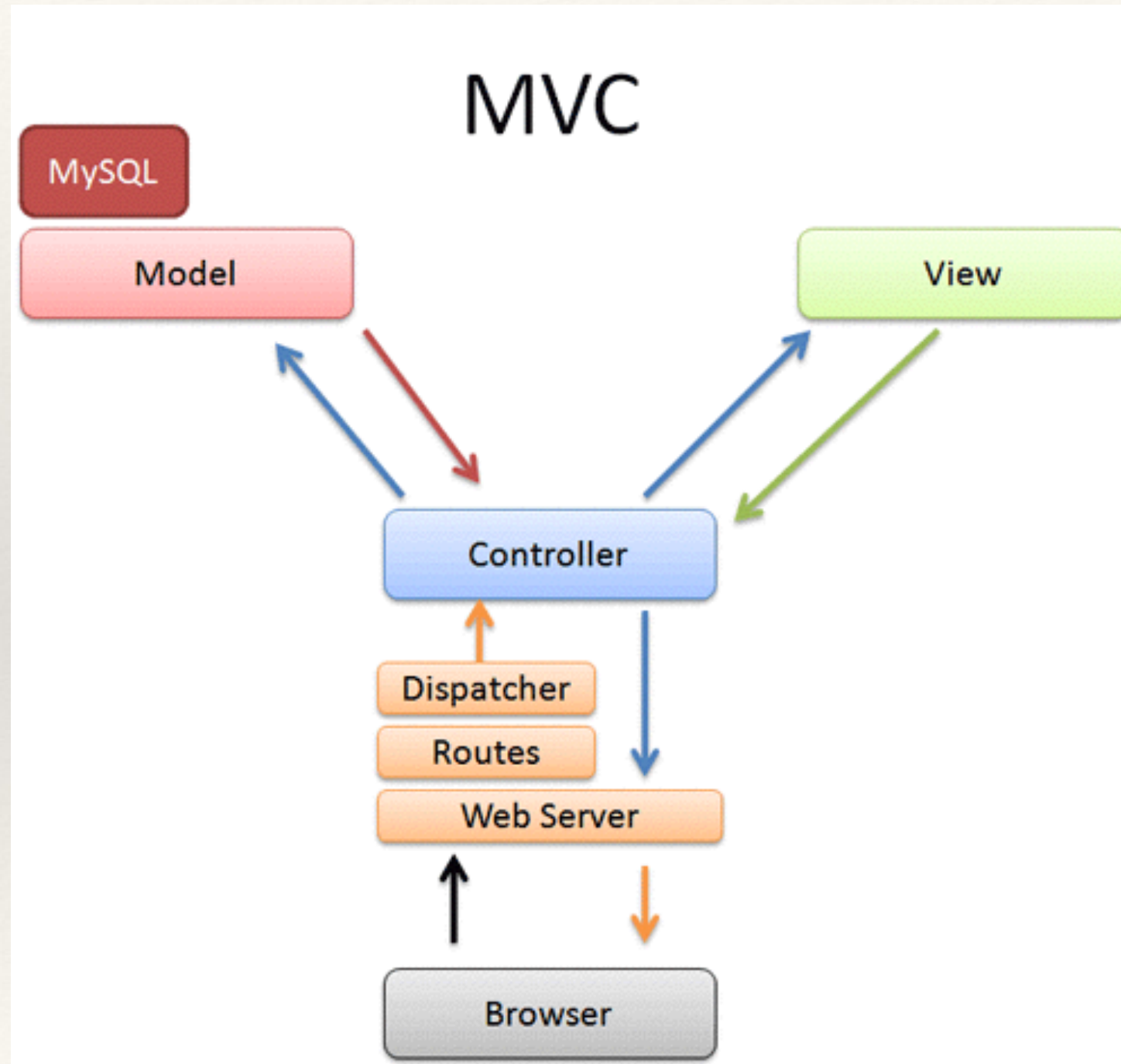# Frameworks

Starting Rails, and the theory behind it

Prerequisites

❖ Rails installed

❖ Text editor installed

Model-View-Controller frameworks

Models

- Models are abstracted representations of the world

- The "data"

Views

- ❖ Views are visual representations of models

- ❖ The "visuals"

Controllers

- ❖ Controllers manipulate data and connect it to views

- ❖ The "brains"

Routers

- ❖ Routers read user-input URLs and connect them to controllers

- ❖ The "traffic policeman"

Databases

❖ Databases hold model data so it can be used again

❖ The "memory"

Now let's create a Rails app!

❖ Specifically, the RailsGirls Diary app…

Creating a Rails application

- In the terminal window:

    - Go to the directory you want your app in

    - Type this:

rails new xxxx

cd xxxx

rails server

- In your browser, go to http://localhost:3000/

- Press control-c in the terminal to stop your app.

What just happened?

❖ Rails created a folder, xxxx, and added files to it

❖ You used "rails server" to start your app

❖ You saw the default homepage for a rails app

The new folder

- Gemfile - libraries

- Gemfile.lock - library versions

- README.rdoc - your description of the application

- app/ - where you put your Ruby code

- config/ - application configuration

- db/ - database

- public/ - all webpages that *don't* get generated by Ruby code

The app folder

- **models**

- **views**

- **controllers**

- mailers

- helpers

- assets

More resources

- http://guides.rubyonrails.org/getting_started.html

- http://railsforzombies.org/

- TeamTreehouse

Example applications

* Spot.us: crowdfunding/ stories

  * http://www.spot.us/

  * http://github.com/spot-us/spot-us

* FatfreeCRM: customer relationship management

  * http://fatfreecrm.com/

  * http://github.com/michaeldv/fat_free_crm

* Spree: shopping

  * http://spreecommerce.com/

  * http://github.com/railsdog/spree

See http://jetpackweb.com/blog/2009/10/14/high-quality-ruby-on-rails-example-applications/