

Backbone.js



Радослав Станков
OpenFest

05/11/2011

Кой съм аз?

@rstankov

<http://rstankov.com>

<http://github.com/rstankov>

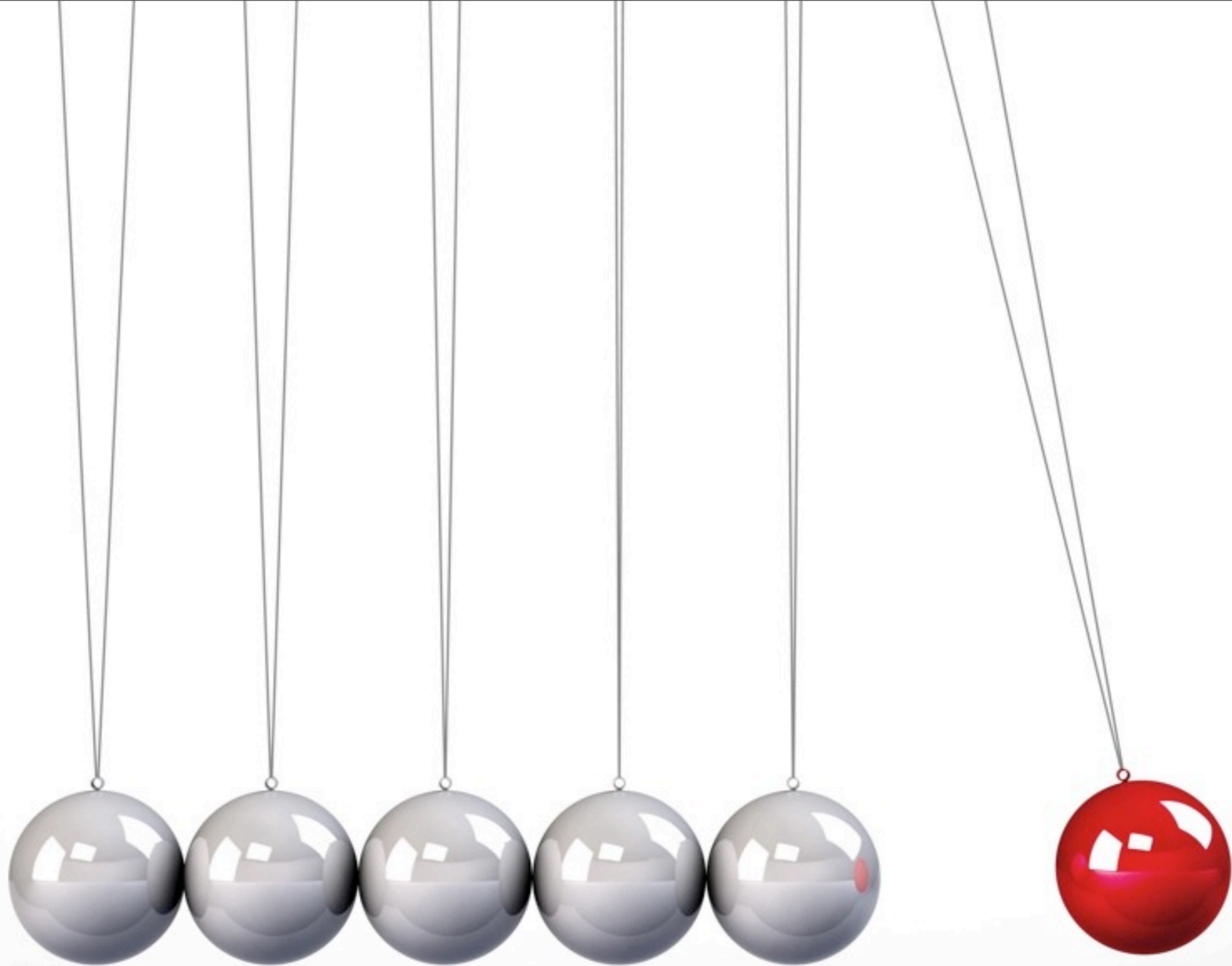






Backbone.js





DocumentCloud

Documents Entitles Search Help

DocumentCloud

All Documents
 Your Documents
 Your Published Documents
 DocumentCloud's Documents


B.P. Deepwater Horizon Oil Spill
 Public documents related to the oil spill, submitted by a variety of newsrooms.
 33 documents, 1 note, 4 collaborators

Commercial Appeal: Ernest Withers ✓
 36 documents, 136 notes

OLC Letters & Memos
 19 documents, 0 notes

The Financial Crisis
 25 documents, 11 notes


+ New Documents + New Project


 **Jeremy Ashkenas**
 Manage Accounts / Log Out


Commercial Appeal: Ernest Withers


project: "Commercial Appeal: Ernest Withers"


Open Edit Projects Analyze Publish


 **Memphis Publishing Company and Marc Perussia v. FBI**
 4 notes


 **Pages From 2009 FOIA Reveal Withers as Informant ME 338**
 2 notes


 **Other Informants**
 4 notes


 **FBI seeks to tie invaders to Black Panthers**
 3 notes

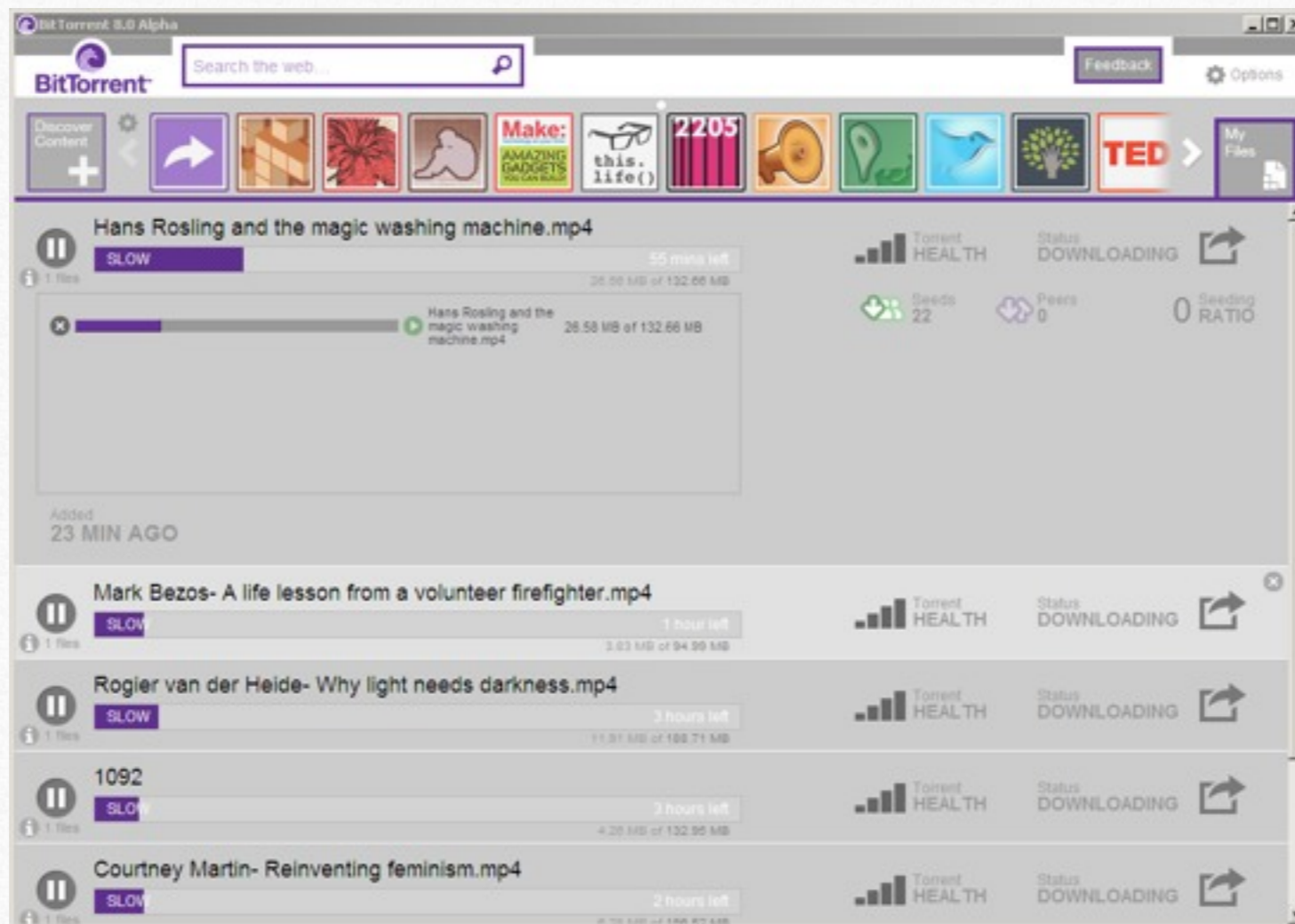
 **FBI targets invaders for prosecution**
 5 notes

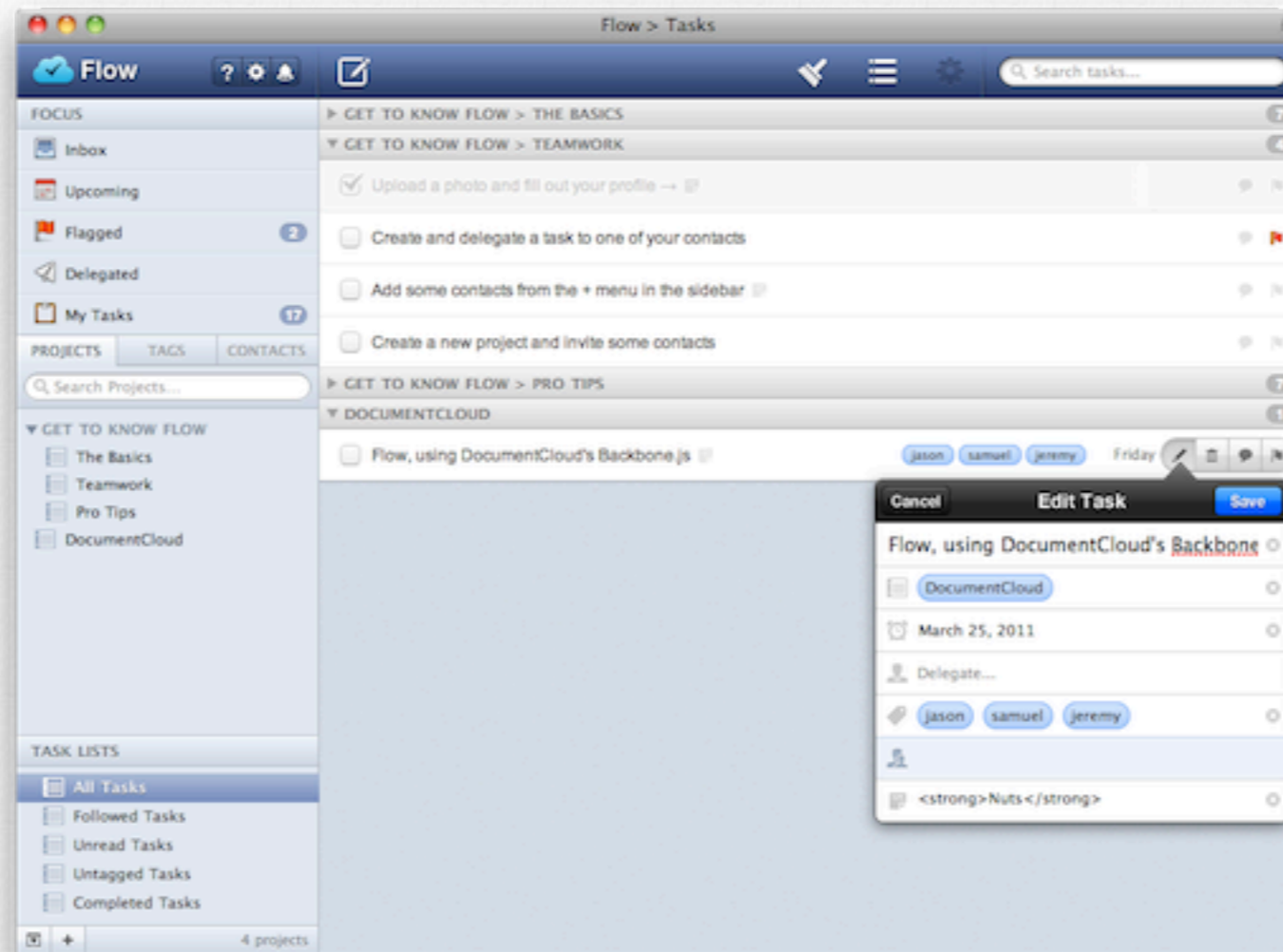
 **Informant draws links between invaders, others**
 4 notes

 **The Invaders and COINTELPRO**
 4 notes

 **Informant Provides Car Tag Number, Other Details**
 2 notes

 **Informant Reports on Political Candidates**
 4 notes





Groupon Now!

Featured Deal All Deals **Now! Deals** Getaways How It Works Sign In Sign Up

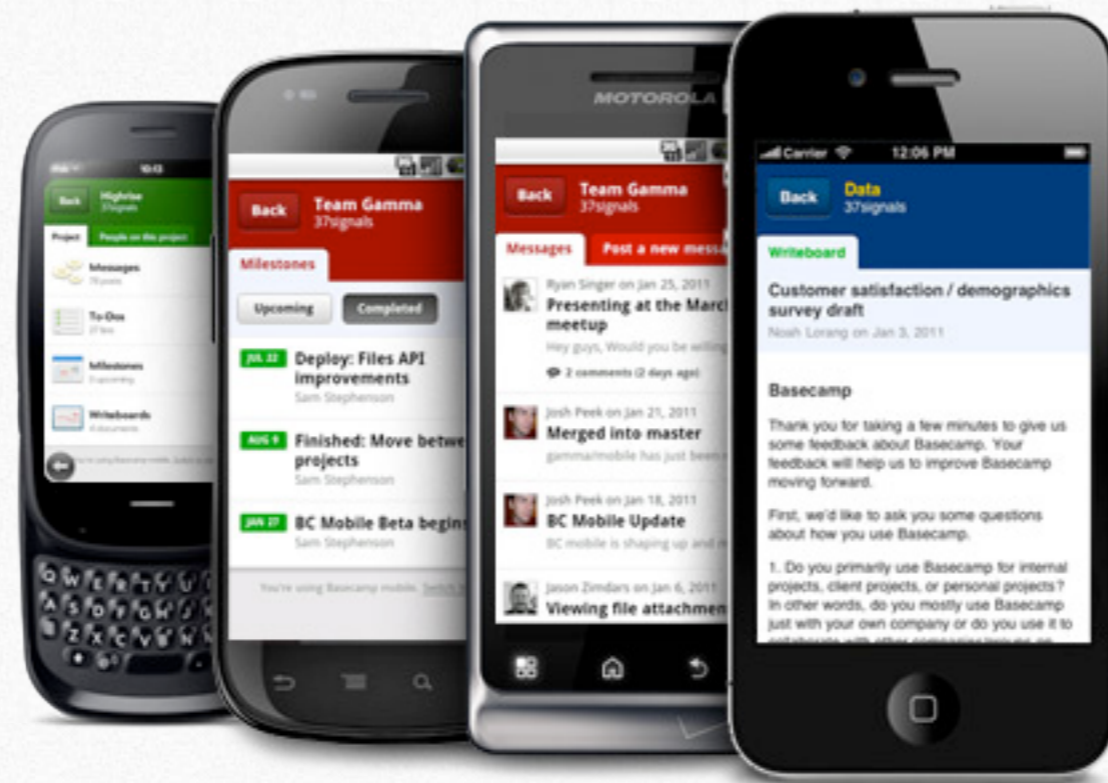
Select category: **Eat Something** Select time: **Today** near **Times Square** **Find Deals**

- A** **\$5 for \$10 at Knave in Le Parker Meridien Lobby**
0.7 miles away ★★★★★ (35)
Redeem today before 4pm
- B** **\$35 for \$70 at 5th Avenue Chocolatier**
0.7 miles away
Redeem today before 7pm
- C** **\$4 for \$7 at Comfort Diner**
0.8 miles away ★★★★★ (57)
Redeem today before 10:30am
- D** **\$18 for \$24 at Opla**
0.9 miles away ★★★★★ (52)
Redeem today before 11am
- E** **\$5 for \$9 at 63bites**
1 mile away ★★★★★ (49)
Redeem today before 1:45pm
- F** **\$6 for \$10 at The Tea Set**
1.5 miles away ★★★★★ (17)
Redeem today before 11pm

Overview Ratings & Fine Print

\$5 for \$9 at 63bites
44% off Homemade Soups, Salads and More
Tweet 0

Buy Now! Redeem Today Between **6am - 1:45pm** or we'll automatically refund you





Trajectory — Trajectory

Trajectory Stories Discussions Settings

Velocity: 4 Completed: 404 stories over 48 iterations

0/4 0% complete 1 Unstarted 0 Completed 2 Started All Design Development

As a user I want to leave comments on the iteration to have a discussion about the week so that I can coordinate with the team. 2 Deliver Stop
Chad Pytel, Design and Development

Todo Site Certification for Credit Card Processing 4 Deliver Stop
Chad Pytel

July 25

Disable add comment button when a comment is being added to prevent double clicking
Unassigned, Development

Bug backbone: Hitting "Deliver" doesn't focus the text input anymore
Chad Pytel, Development

Bug Discussion layout breaks with many stories assigned to discussion. See attachment. 2
Unassigned, Development

Todo Put "Accept" button on the left, "Reject" on the right. 3
Unassigned

Bug Closing points estimation popup is not working. Popup should close when clicking outside the popup. 1
Unassigned, Development

Bug As a user, when I delete a story, the iterations should be recalculated. 1
Unassigned

Bug When I click on the completed stories link on the top right (expanding prior iteration's stories), and then click on the completed button on this iteration's toolbar, the window scrolls to the top. It should stay there so that I can easily see stories completed on this iteration. 1
Unassigned

Bug Backbone links aren't styled fancifully 1
Unassigned, Development

August 01

Backbone.js



Browser

Controller

Model

View

Сървър

Browser



Controller

Model

View

Сървър

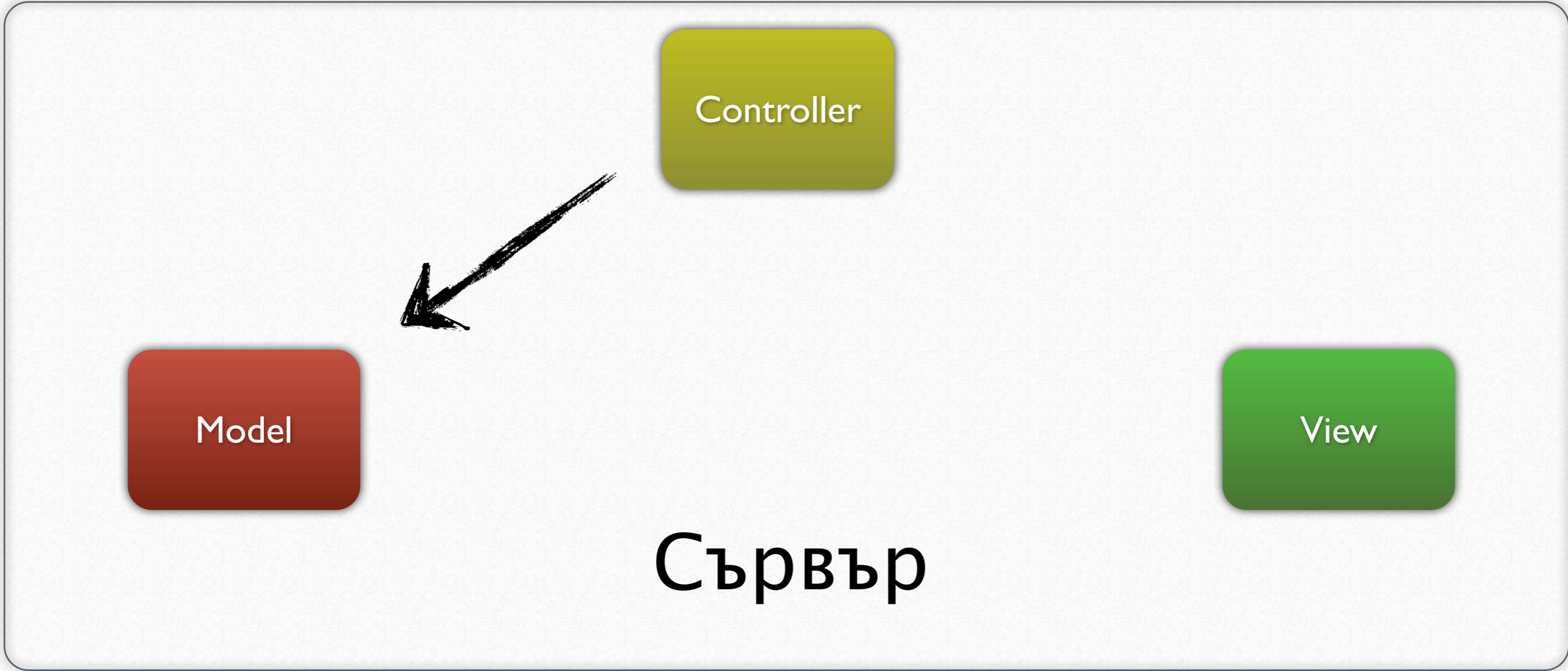
Browser

Controller

Model

View

Сървър



Browser

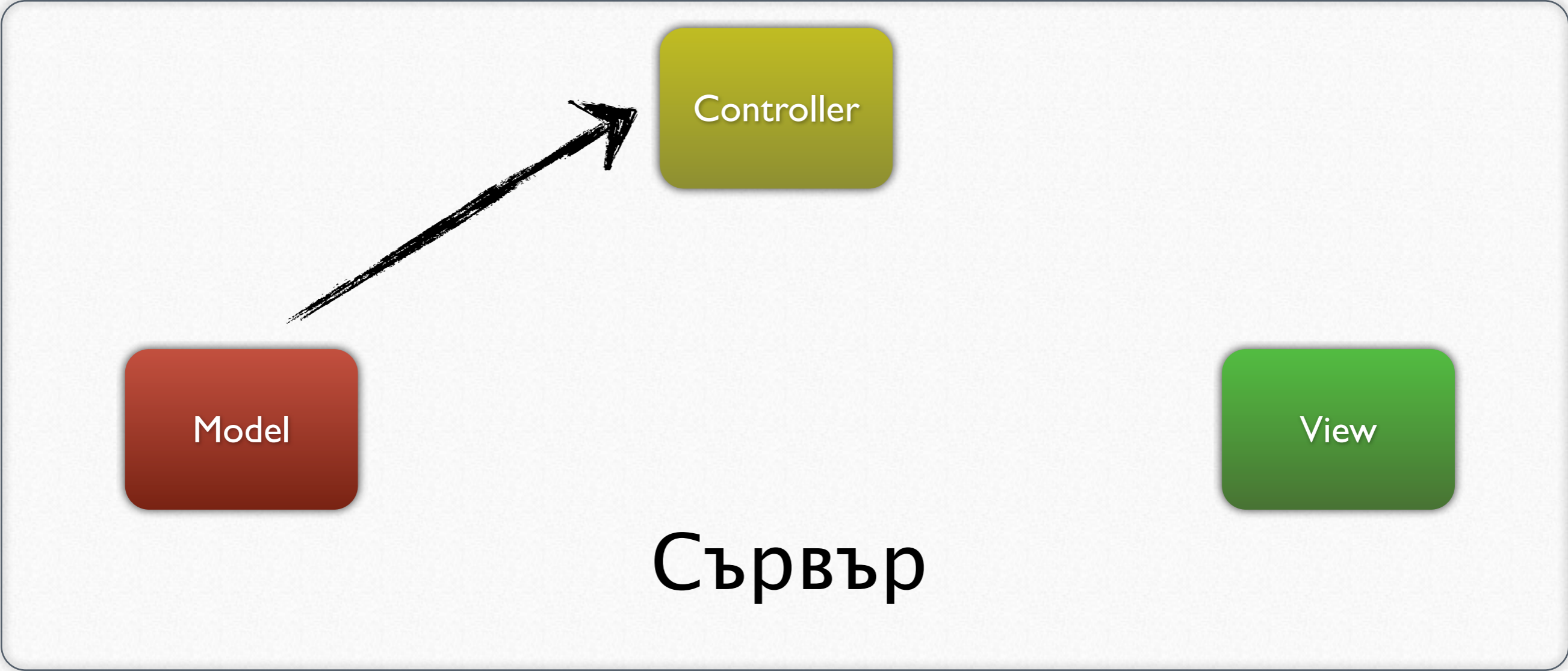
Controller

Model

View

Сървър

Browser



Сървър



Browser

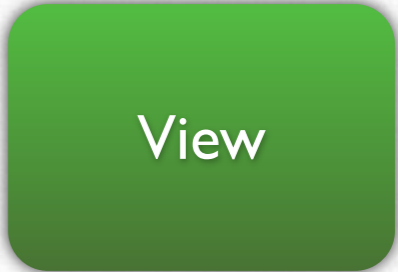
Controller

Model

View

Сървър





Сървър

Browser

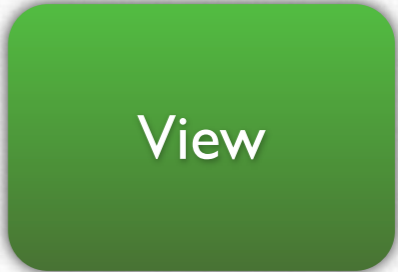
Controller

Model

View

Сървър





Сървър

Browser

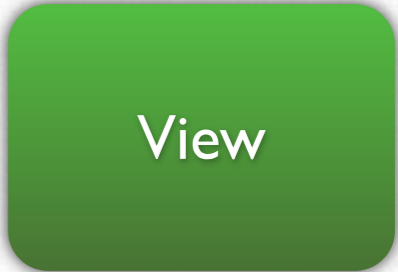


Controller

Model

View

Сървър



Сървър

Browser



Browser



Browser



Browser



Browser



Browser



Browser



Browser



Browser



Backbone.js



Backbone.Events

```
var object = {};  
  
$.extend(object, Backbone.Events);  
  
object.bind('eventName', function() {  
  console.log('1');  
});  
  
object.bind('eventName', function() {  
  console.log('2');  
});  
  
object.trigger('eventName');  
  
// prints '1' and '2'
```

```
var object = {};
```

```
$.extend(object, Backbone.Events);
```

```
object.bind('eventName', function() {  
  console.log('1');  
});
```

```
object.bind('eventName', function() {  
  console.log('2');  
});
```

```
object.trigger('eventName');
```

```
// prints '1' and '2'
```

```
var object = {};
```

```
$.extend(object, Backbone.Events);
```

```
object.bind('eventName', function() {  
  console.log('1');  
});
```

```
object.bind('eventName', function() {  
  console.log('2');  
});
```

```
object.trigger('eventName');
```

```
// prints '1' and '2'
```

```
var object = {};  
$.extend(object, Backbone.Events);  
object.bind('eventName', function() {  
  console.log('1');  
});
```

```
object.bind('eventName', function() {  
  console.log('2');  
});
```

```
object.trigger('eventName');
```

```
// prints '1' and '2'
```

```
var object = {};
```

```
$.extend(object, Backbone.Events);
```

```
object.bind('eventName', function() {  
  console.log('1');  
});
```

```
object.bind('eventName', function() {  
  console.log('2');  
});
```

```
object.trigger('eventName');
```

```
// prints '1' and '2'
```

Backbone.js



Backbone.Model

```
var Person = Backbone.Model.extend({  
  initialize: function(){  
    console.log("I'm alive!");  
  }  
});
```

```
new Person();
```

```
var Person = Backbone.Model.extend({
  initialize: function(){
    console.log("I'm alive!");
  }
});
```

```
new Person();
```

```
var Person = Backbone.Model.extend({
  initialize: function(){
    console.log("I'm alive!");
  }
});
```

```
new Person();
```

```
var Person = Backbone.Model.extend({});  
  
var me = new Person({name: 'Radoslav'});  
  
me.get('name'); // Radoslav  
me.set({lastName: 'Stankov'});  
me.get('lastName'); // Stankov
```

```
var Person = Backbone.Model.extend({});
```

```
var me = new Person({name: 'Radoslav'});
```

```
me.get('name'); // Radoslav
```

```
me.set({lastName: 'Stankov'});
```

```
me.get('lastName'); // Stankov
```

```
var Person = Backbone.Model.extend({});
```

```
var me = new Person({name: 'Radoslav'});
```

```
me.get('name'); // Radoslav
```

```
me.set({lastName: 'Stankov'});
```

```
me.get('lastName'); // Stankov
```

```
var Person = Backbone.Model.extend({});  
  
var me = new Person({name: 'Radoslav'});  
  
me.get('name'); // Radoslav  
me.set({lastName: 'Stankov'});  
me.get('lastName'); // Stankov
```

```
var Person = Backbone.Model.extend({});
```

```
var me = new Person({name: 'Radoslav'});
```

```
me.get('name'); // Radoslav  
me.set({lastName: 'Stankov'});  
me.get('lastName'); // Stankov
```

```
var Person = Backbone.Model.extend({
  defaults: {
    name: 'John',
    lastName: 'Doe',
  }
});
```

```
var me = new Person();
me.get('name'); // John
me.get('lastName'); // Doe
```

```
var Person = Backbone.Model.extend({  
  defaults: {  
    name: 'John',  
    lastName: 'Doe',  
  }  
});
```

```
var me = new Person();  
me.get('name'); // John  
me.get('lastName'); // Doe
```

```
var Person = Backbone.Model.extend({
  defaults: {
    name: 'John',
    lastName: 'Doe',
  }
});
```

```
var me = new Person();
me.get('name'); // John
me.get('lastName'); // Doe
```

```
var Person = Backbone.Model.extend({
  defaults: {
    name: 'John',
    lastName: 'Doe',
  }
});
```

```
var me = new Person();
me.get('name'); // John
me.get('lastName'); // Doe
```

```
var Calculator = Backbone.Model.extend({
  defaults: {
    value: 0
  },
  value: function(){
    return this.get('value');
  },
  sum: function(value){
    this.set({value: value + this.get('value')});
  },
  reset: function(){
    this.set({value: 0});
  }
});
```

```
var Calculator = Backbone.Model.extend({
  defaults: {
    value: 0
  },
  value: function(){
    return this.get('value');
  },
  sum: function(value){
    this.set({value: value + this.get('value')});
  },
  reset: function(){
    this.set({value: 0});
  }
});
```

```
var Calculator = Backbone.Model.extend({
  defaults: {
    value: 0
  },
  value: function(){
    return this.get('value');
  },
  sum: function(value){
    this.set({value: value + this.get('value')});
  },
  reset: function(){
    this.set({value: 0});
  }
});
```

```
var Calculator = Backbone.Model.extend({
  defaults: {
    value: 0
  },
  value: function(){
    return this.get('value');
  },
  sum: function(value){
    this.set({value: value + this.get('value')});
  },
  reset: function(){
    this.set({value: 0});
  }
});
```

```
var Calculator = Backbone.Model.extend({
  defaults: {
    value: 0
  },
  value: function(){
    return this.get('value');
  },
  sum: function(value){
    this.set({value: value + this.get('value')});
  },
  reset: function(){
    this.set({value: 0});
  }
});
```

```
var cal = new Calculator();

cal.bind('change:value', function(model, value){
  console.log(value);
});

cal.bind('change', function(model){
  console.log(model.get('value'));
});

cal.bind('all', function(eventName) {
  console.log('I see every thing!', eventName);
});
```

```
var cal = new Calculator();
```

```
cal.bind('change:value', function(model, value){  
  console.log(value);  
});
```

```
cal.bind('change', function(model){  
  console.log(model.get('value'));  
});
```

```
cal.bind('all', function(eventName) {  
  console.log('I see every thing!', eventName);  
});
```

```
var cal = new Calculator();
```

```
cal.bind('change:value', function(model, value){  
    console.log(value);  
});
```

```
cal.bind('change', function(model){  
    console.log(model.get('value'));  
});
```

```
cal.bind('all', function(eventName) {  
    console.log('I see every thing!', eventName);  
});
```

```
var cal = new Calculator();
```

```
cal.bind('change:value', function(model, value){  
    console.log(value);  
});
```

```
cal.bind('change', function(model){  
    console.log(model.get('value'));  
});
```

```
cal.bind('all', function(eventName) {  
    console.log('I see every thing!', eventName);  
});
```

```
var cal = new Calculator();  
  
cal.bind('myEvent', function(){  
    console.log('KaBoom....');  
});  
  
cal.trigger('myEvent');
```

```
var Product = Backbone.Model.extend({  
  urlRoot: '/products'  
});
```

```
var chair = new Product({  
  name: 'chair',  
  price: 10  
});
```

```
chair.save();
```

```
// POST /products
```

```
var Product = Backbone.Model.extend({
  url: function(){
    return '/products/' + (this.isNew() ? '' : this.id);
  }
});
```

```
var chair = new Product({
  id: 5,
  name: 'chair',
  price: 10
});
```

```
chair.save();
```

```
// PUT /products/1
```

<http://documentcloud.github.com/backbone/#Model-save>

И още...

- validate
- escape
- has
- unset
- clear
- hasChanged
- changedAttributes
- previousAttributes
- fetch
- toJSON
- clone

Backbone.js



Backbone.View

Todos

What needs to be done?

- Model
- Collection
- View
- Router

4 items left.

div.content 480px×291px

```
<!DOCTYPE html>
<html>
  <head>...</head>
  <style type="text/css">...</style>
  <style type="text/css" style="display: none !important; ">...</style>
  <body>
    <!-- Todo App Interface -->
    <div id="todoapp">
      <div class="title">...</div>
      <div class="content">
        <div id="create-todo">
          <input id="new-todo" placeholder="What needs to be done?" type="text">
          <span class="ui-tooltip-top" style="display: none; ">Press Enter to save this task</span>
        </div>
        <div id="todos">...</div>
        <div id="todo-stats">...</div>
      </div>
    <!-- Templates -->
    <script type="text/template" id="item-template">...</script>
    <script type="text/template" id="stats-template">...</script>
  </body>
```

```
var UserNameView = Backbone.View.extend({
  tagName: 'input',
  className: 'string optional',
  id: 'user-name',
  attributes: {
    type: 'string',
    name: 'user[name]'
  }
});
```

```
var userName = new UserNameView();
```

```
console.log(userName.el);
```

```
<input type="string" name="user[name]" id="user-name" class="string optional">
```

```
var UserNameView = Backbone.View.extend({  
  tagName: 'input',  
  className: 'string optional',  
  id: 'user-name',  
  attributes: {  
    type: 'string',  
    name: 'user[name]'  
  }  
});
```

```
var userName = new UserNameView();
```

```
console.log(userName.el);
```

```
<input type="string" name="user[name]" id="user-name" class="string optional">
```

```
var UserNameView = Backbone.View.extend({
  tagName: 'input',
  className: 'string optional',
  id: 'user-name',
  attributes: {
    type: 'string',
    name: 'user[name]'
  }
});
```

```
var userName = new UserNameView();
```

```
console.log(userName.el);
```

```
<input type="string" name="user[name]" id="user-name" class="string optional">
```

```
var UserNameView = Backbone.View.extend({
  tagName: 'input',
  className: 'string optional',
  id: 'user-name',
  attributes: {
    type: 'string',
    name: 'user[name]'
  }
});
```

```
var userName = new UserNameView();

console.log(userName.el);
```

```
<input type="string" name="user[name]" id="user-name" class="string optional">
```

```
var UserNameView = Backbone.View.extend({
  tagName: 'input',
  className: 'string optional',
  id: 'user-name',
  attributes: {
    type: 'string',
    name: 'user[name]'
  }
});
```

```
var userName = new UserNameView();
```

```
console.log(userName.el);
```

```
<input type="string" name="user[name]" id="user-name" class="string optional">
```

```
var UserNameView = Backbone.View.extend({
  tagName: 'input',
  className: 'string optional',
  id: 'user-name',
  attributes: {
    type: 'string',
    name: 'user[name]'
  }
});
```

```
var userName = new UserNameView();
```

```
console.log(userName.el);
```

```
<input type="string" name="user[name]" id="user-name" class="string optional">
```

```
var UserNameView = Backbone.View.extend({
  tagName: 'input',
  className: 'string optional',
  id: 'user-name',
  attributes: {
    type: 'string',
    name: 'user[name]'
  }
});
```

```
var userName = new UserNameView();
```

```
console.log(userName.el);
```

```
<input type="string" name="user[name]" id="user-name" class="string optional">
```

```
var UsersListView = Backbone.View.extend({  
  el: '#users-list'  
});
```

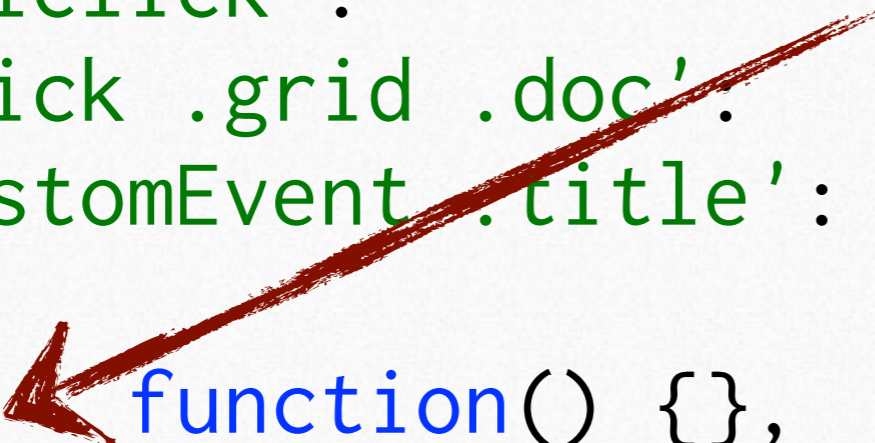
```
var userList = new UsersListView();
```

```
console.log(userList.el);
```

```
var EditBoxView = Backbone.View.extend({});  
  
var element = $('#edit-box-view').get(0),  
    editBox = new EditBoxView({el: element});  
  
console.log(editBox.el === element);
```

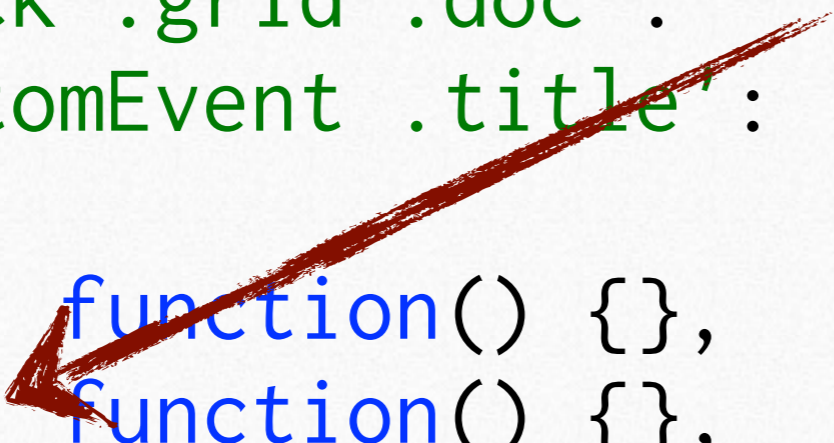
```
var DocumentView = Backbone.View.extend({
  events: {
    'dblclick':      'open',
    'click .grid .doc': 'select',
    'customEvent .title': 'custom'
  },
  open:  function() {},
  select: function() {},
  custom: function() {}
});
```

```
var DocumentView = Backbone.View.extend({
  events: {
    'dblclick': 'open',
    'click .grid .doc': 'select',
    'customEvent .title': 'custom'
  },
  open: function() {},
  select: function() {},
  custom: function() {}
});
```



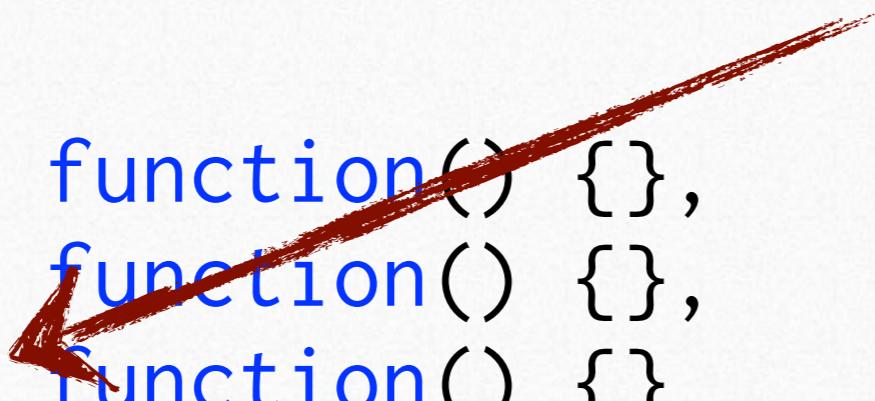
```
var DocumentView = Backbone.View.extend({
  events: {
    'dblclick':      'open',
    'click .grid .doc': 'select',
    'customEvent .title': 'custom'
  },
  open:  function() {},
  select: function() {},
  custom: function() {}
});
```

```
var DocumentView = Backbone.View.extend({
  events: {
    'dblclick': 'open',
    'click .grid .doc': 'select',
    'customEvent .title': 'custom'
  },
  open: function() {},
  select: function() {},
  custom: function() {}
});
```



```
var DocumentView = Backbone.View.extend({
  events: {
    'dblclick':      'open',
    'click .grid .doc': 'select',
    'customEvent .title': 'custom'
  },
  open:  function() {},
  select: function() {},
  custom: function() {}
});
```

```
var DocumentView = Backbone.View.extend({
  events: {
    'dblclick':      'open',
    'click .grid .doc': 'select',
    'customEvent .title': 'custom'
  },
  open:  function() {},
  select: function() {},
  custom: function() {}
});
```



```
var DocumentView = Backbone.View.extend({
  events: {
    'dblclick':      'open',
    'click .grid .doc': 'select',
    'customEvent .title': 'custom'
  },
  open:  function() {},
  select: function() {},
  custom: function() {}
});
```

```
<script type="text/template" id="news">  
  <h1><%= title %></h1>  
  <time><%= created_at %></time>  
  <p><%= text %></p>  
</script>
```

```
var NewsView = Backbone.View.extend({
  template: _.template($('#news').html()),
  render: function() {
    this.el.innerHTML = this.template(this.model);
    return this;
  }
});
```

```
var view = new NewsView({  
  model: {  
    title: "News Title",  
    created_at: "Today",  
    text: "Long text"  
  }  
});
```

```
document.body.appendChild(view.render().el);
```

```
<script type="text/template" id="news">  
  <h1><%= title %></h1>  
  <time><%= created_at %></time>  
  <p><%= text %></p>  
</script>
```

```
<div>  
  <h1>News Title</h1>  
  <time>Today</time>  
  <p>Long text</p>  
</div>
```

View

Model

DOM

Model

View

DOM



View

Model

DOM

View



Model

DOM

View

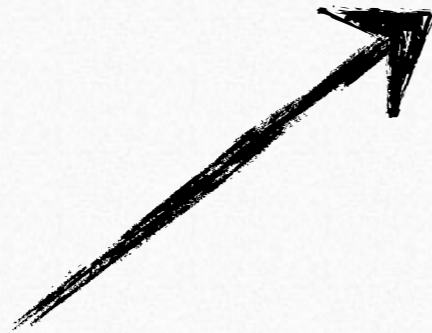
Model

DOM

Model

View

DOM



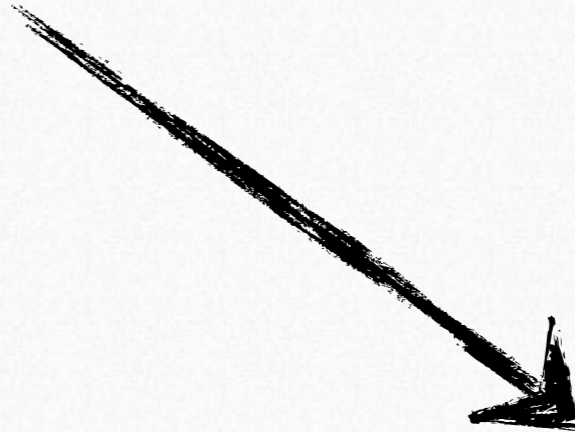
View

Model

DOM

Model

View



DOM

View

Model

DOM

View

Model

DOM

View

Model

DOM

View 2

Model

View

View 2

DOM

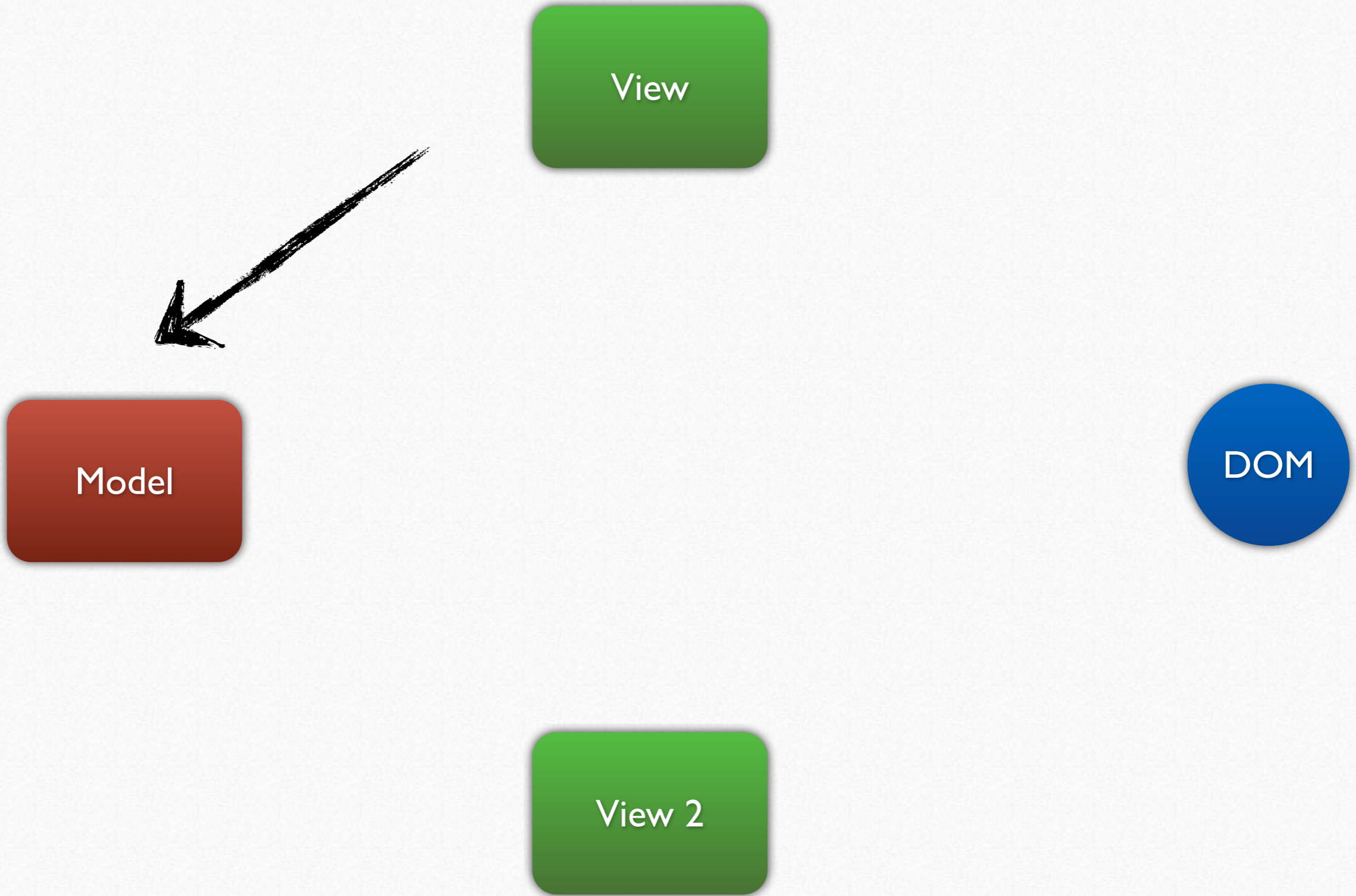


View

Model

DOM

View 2

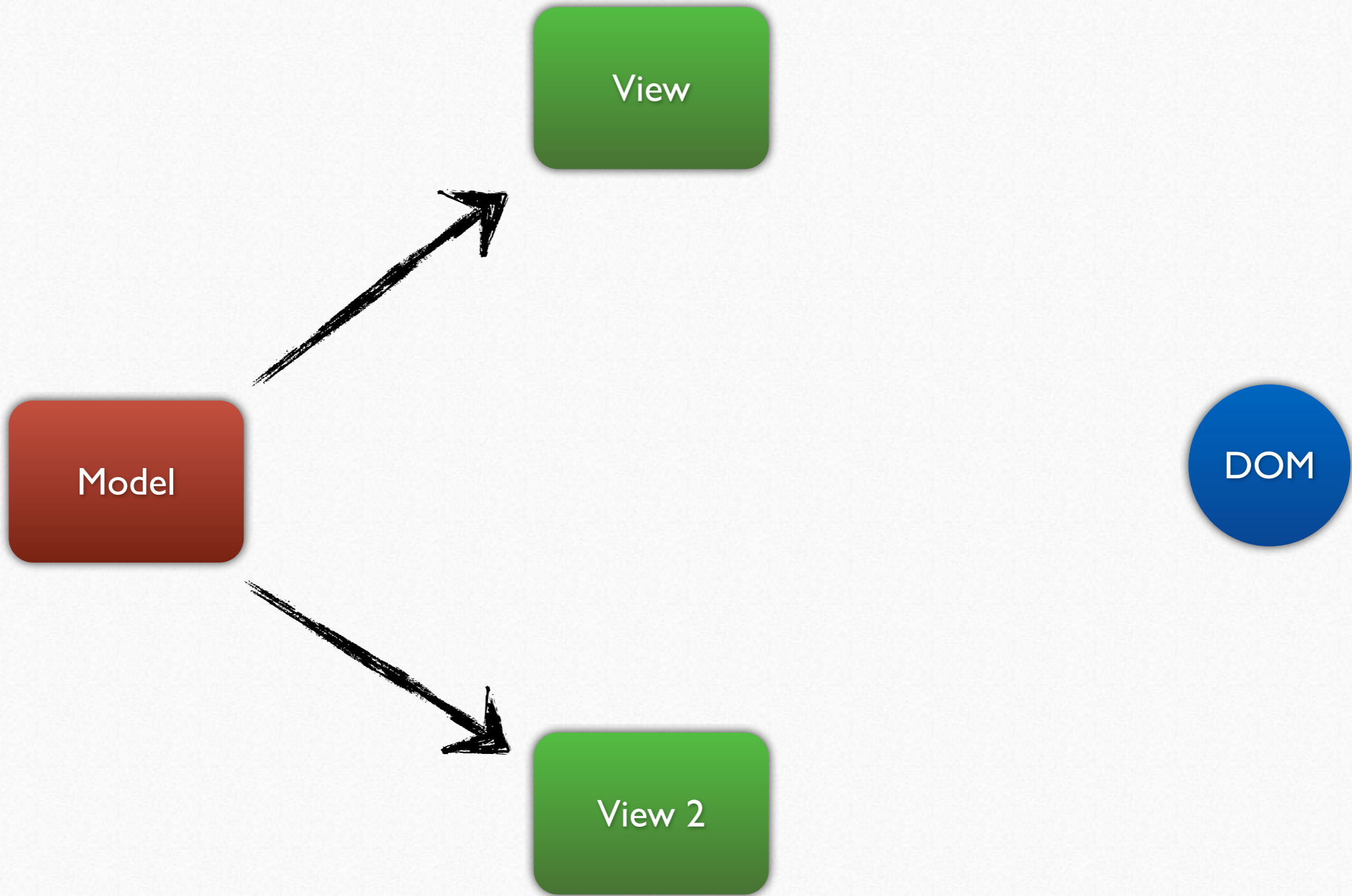


View

Model

DOM

View 2



View

Model

DOM

View 2

Model

View

View 2

DOM



View

Model

DOM

View 2

View

Model

DOM

View 2

Model

View

View 2

DOM

Model

View

View 2

View 3

DOM

Model

View

View 2

View 3

View 4

DOM

Model

View

View 2

View 3

View 4

View .. N

DOM

Model

View

View 2

View 3

View 4

View .. N



DOM

Model

View

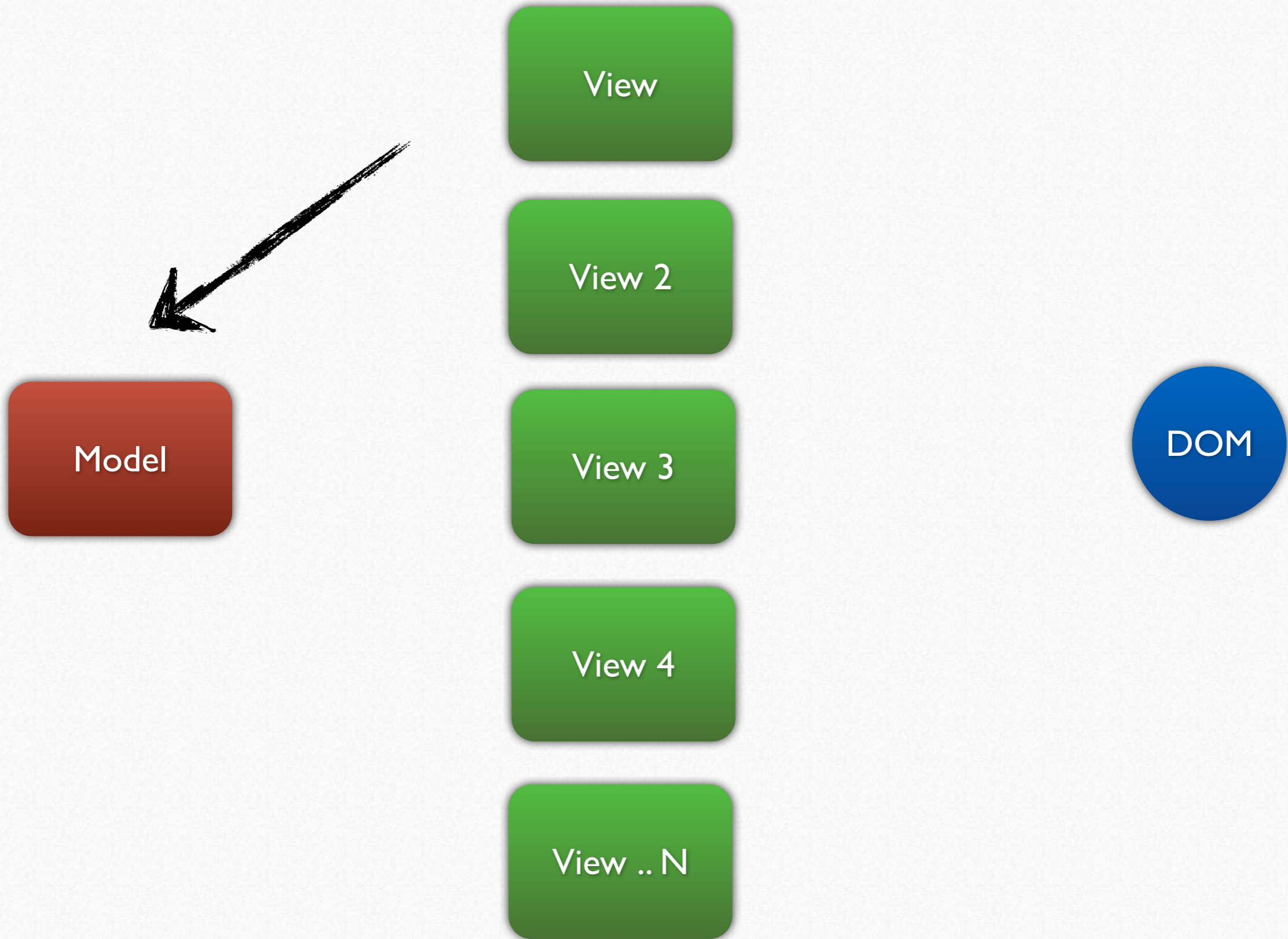
View 2

View 3

View 4

View .. N

DOM



Model

View

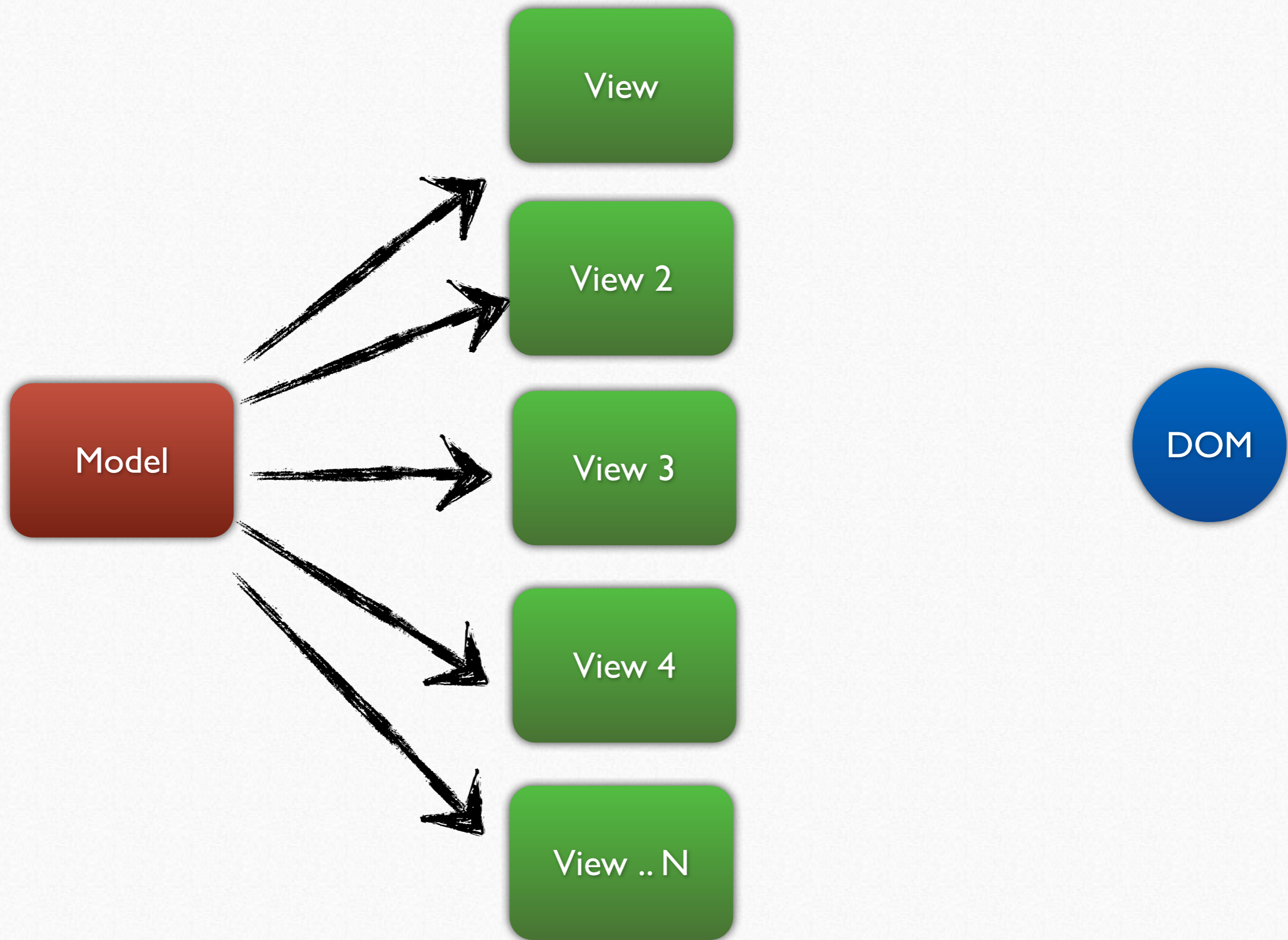
View 2

View 3

View 4

View .. N

DOM



Model

View

View 2

View 3

View 4

View .. N

DOM

Model

View

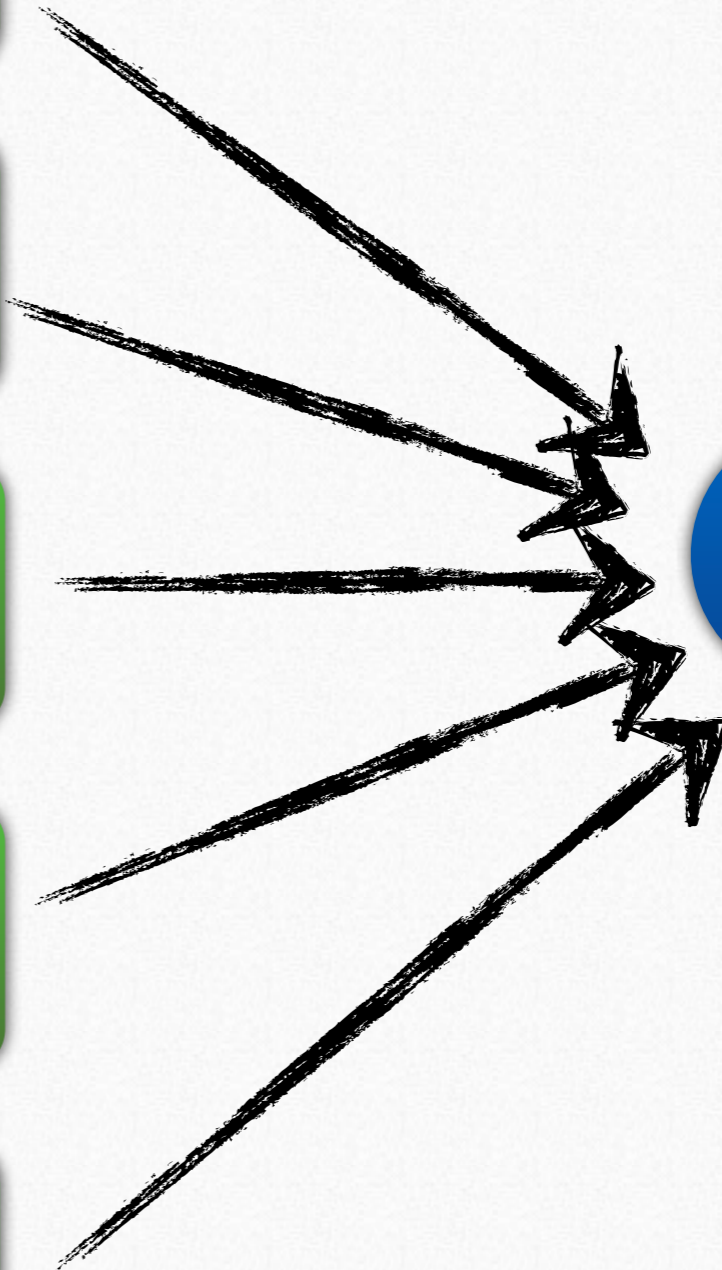
View 2

View 3

View 4

View .. N

DOM



Model

View

View 2

View 3

View 4

View .. N

DOM

Demo



```
var Calculator = Backbone.Model.extend({
  defaults: {
    value: 0
  },
  increment: function() {
    this.set({'value': this.get('value') + 1});
  },
  decrement: function() {
    this.set({'value': this.get('value') - 1});
  },
  getValue: function() {
    return this.get('value');
  }
});
```

```
var ButtonsView = Backbone.View.extend({
  events: {
    'click .plus': 'plus',
    'click .minus': 'minus'
  },
  plus: function() {
    this.model.increment();
  },
  minus: function() {
    this.model.decrement();
  }
});
```

```
var DisplayView = Backbone.View.extend({
  initialize: function() {
    this.model.bind('change:value', this.render, this);
    this.render();
  },
  render: function() {
    this.el.innerHTML = this.model.getValue();
    return this;
  }
});
```

```
var cal = new Calculator();  
  
new ButtonsView({model: cal, el: '.buttons'});  
new DisplayView({model: cal, el: '.display'});
```

Backbone.js



Backbone.Collection

```
var ProductsCollection = Backbone.Collection.extend({  
  model: Product  
});
```

```
var products = new ProductsCollection();
```

```
products.fetch();
```

```
products.bind('reset', function(list) {  
  console.log('Loaded', list.length, 'records');  
});
```

```
products.bind('add', function(model) {  
  console.log('new product added');  
});
```

```
products.bind('remove', function(model) {  
  console.log('item product removed');  
});
```

И още...

- Underscore Methods
- add / remove / at
- sort / comparator
- reset
- create
- url
- toJSON

Backbone.js



Backbone.Router



<http://example.com/#/page/backbone-demo>

```
var AppRouter = Backbone.Router.extend({
  routes: {
    'pages': 'index',
    'pages/search/:q': 'search',
    'pages/:id': 'show'
  },
  initialize: function() {
    console.log('initialize');
  },
  index: function() { /* code */ },
  search: function(query) { /* code */ },
  show: function(id) { /* code */ }
});
```

```
var app = new AppRouter();
```

```
Backbone.history.start();
```

```
var AppRouter = Backbone.Router.extend({
  routes: {
    'pages': 'index',
    'pages/search/:q': 'search',
    'pages/:id': 'show'
  },
  initialize: function() {
    console.log('initialize');
  },
  index: function() { /* code */ },
  search: function(query) { /* code */ },
  show: function(id) { /* code */ }
});
```

```
var app = new AppRouter();
```

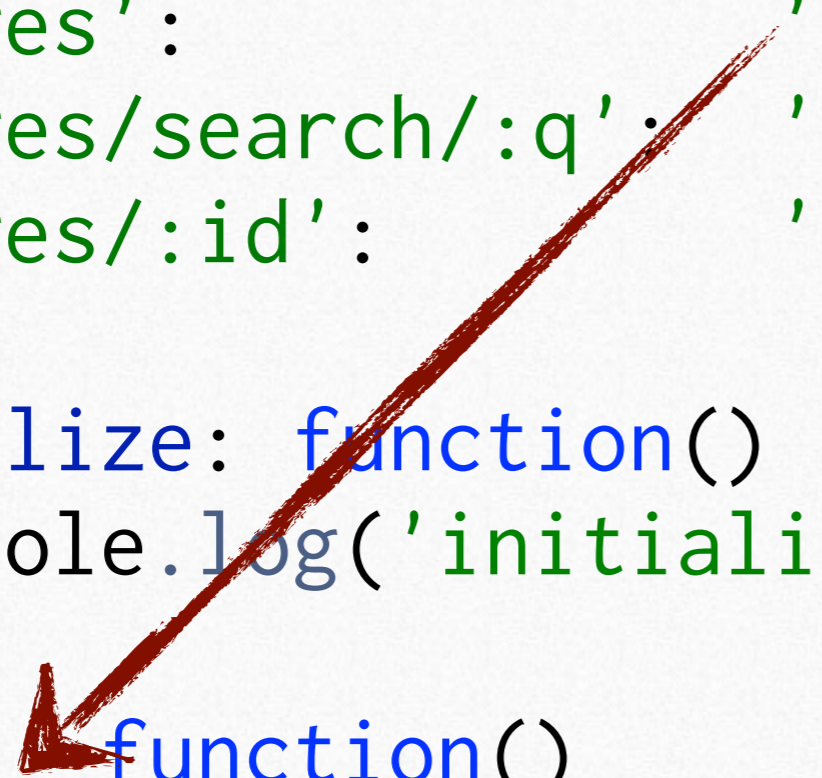
```
Backbone.history.start();
```

```
var AppRouter = Backbone.Router.extend({
  routes: {
    'pages': 'index',
    'pages/search/:q': 'search',
    'pages/:id': 'show'
  },
  initialize: function() {
    console.log('initialize');
  },
  index: function() { /* code */ },
  search: function(query) { /* code */ },
  show: function(id) { /* code */ }
});
```

```
var app = new AppRouter();
```

```
Backbone.history.start();
```

```
var AppRouter = Backbone.Router.extend({
  routes: {
    'pages': 'index',
    'pages/search/:q': 'search',
    'pages/:id': 'show'
  },
  initialize: function() {
    console.log('initialize');
  },
  index: function() { /* code */ },
  search: function(query) { /* code */ },
  show: function(id) { /* code */ }
});
```



```
var app = new AppRouter();
```

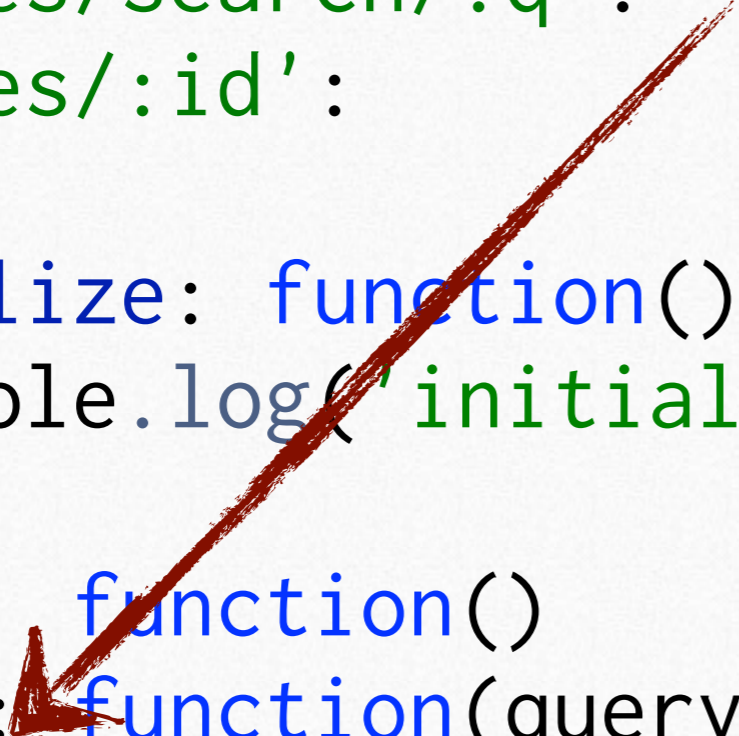
```
Backbone.history.start();
```

```
var AppRouter = Backbone.Router.extend({
  routes: {
    'pages': 'index',
    'pages/search/:q': 'search',
    'pages/:id': 'show'
  },
  initialize: function() {
    console.log('initialize');
  },
  index: function() { /* code */ },
  search: function(query) { /* code */ },
  show: function(id) { /* code */ }
});
```

```
var app = new AppRouter();
```

```
Backbone.history.start();
```

```
var AppRouter = Backbone.Router.extend({
  routes: {
    'pages': 'index',
    'pages/search/:q': 'search',
    'pages/:id': 'show'
  },
  initialize: function() {
    console.log('initialize');
  },
  index: function() { /* code */ },
  search: function(query) { /* code */ },
  show: function(id) { /* code */ }
});
```



```
var app = new AppRouter();
```

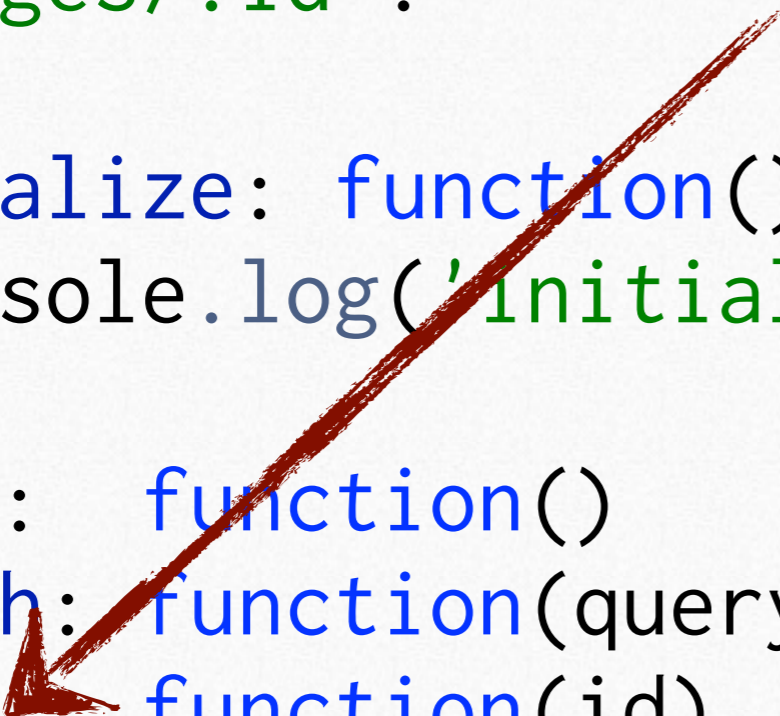
```
Backbone.history.start();
```

```
var AppRouter = Backbone.Router.extend({
  routes: {
    'pages': 'index',
    'pages/search/:q': 'search',
    'pages/:id': 'show'
  },
  initialize: function() {
    console.log('initialize');
  },
  index: function() { /* code */ },
  search: function(query) { /* code */ },
  show: function(id) { /* code */ }
});
```

```
var app = new AppRouter();
```

```
Backbone.history.start();
```

```
var AppRouter = Backbone.Router.extend({
  routes: {
    'pages': 'index',
    'pages/search/:q': 'search',
    'pages/:id': 'show'
  },
  initialize: function() {
    console.log('initialize');
  },
  index: function() { /* code */ },
  search: function(query) { /* code */ },
  show: function(id) { /* code */ }
});
```



```
var app = new AppRouter();
```

```
Backbone.history.start();
```

```
var AppRouter = Backbone.Router.extend({
  routes: {
    'pages': 'index',
    'pages/search/:q': 'search',
    'pages/:id': 'show'
  },
  initialize: function() {
    console.log('initialize');
  },
  index: function() { /* code */ },
  search: function(query) { /* code */ },
  show: function(id) { /* code */ }
});
```

```
var app = new AppRouter();
```

```
Backbone.history.start();
```

```
var AppRouter = Backbone.Router.extend({
  routes: {
    'pages': 'index',
    'pages/search/:q': 'search',
    'pages/:id': 'show'
  },
  initialize: function() {
    console.log('initialize');
  },
  index: function() { /* code */ },
  search: function(query) { /* code */ },
  show: function(id) { /* code */ }
});
```

```
var app = new AppRouter();

Backbone.history.start();
```

```
var AppRouter = Backbone.Router.extend({
  routes: {
    'pages': 'index',
    'pages/search/:q': 'search',
    'pages/:id': 'show'
  },
  initialize: function() {
    console.log('initialize');
  },
  index: function() { /* code */ },
  search: function(query) { /* code */ },
  show: function(id) { /* code */ }
});
```

```
var app = new AppRouter();
```

```
Backbone.history.start();
```

```
app.navigate('pages', true);  
app.navigate('pages/search/title', true);  
app.navigate('pages/3', true);
```

site.com/path#pages

site.com/path#pages/search/title

site.com/path#pages/3

HTML



```
Backbone.history.start({pushState: true});
```

site.com/path#pages

site.com/path#pages/search/title

site.com/path#pages/3

site.com/path/pages

site.com/path/pages/search/title

site.com/path/pages/3

Backbone.js





@jimweirich

Jim Weirich

OH: "It is half-assed, but it the right half
of the ass."

25 Jan via [Twitter for iPad](#)

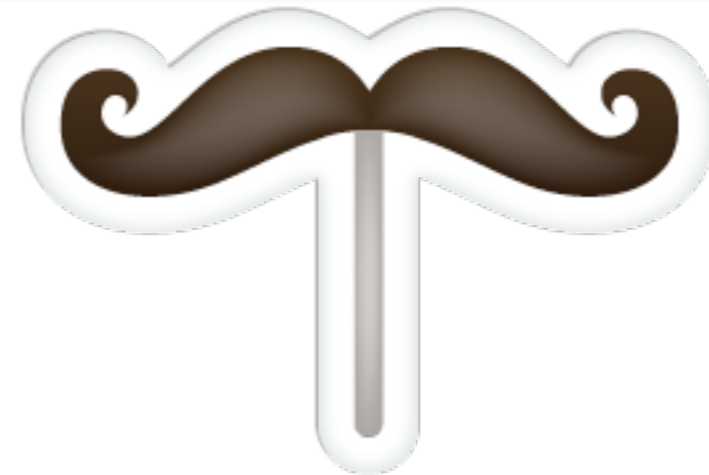






handlebars





Backbone.js



Недостатъци

Алтернативи

Алтернативи



Алтернативи

Spine



Алтернативи



Spine



Въпроси?

Благодаря за вниманието

@rstankov