

A framework for creating ambitious web Applications <a href="http://emberjs.com">http://emberjs.com</a>

### YEHUDA KATZ

- Tilde Inc
- Ember.js
- Jsonapi.org
- Ruby on Rails
- jQuery Project
- Strobe Inc.
- Engine Yard
- Google



# TOM DALE



- Tilde Inc.
- Apple Inc.
- Strobe Inc.

#### GROUPON

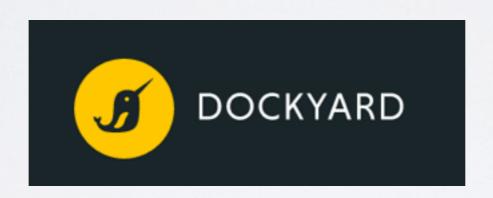


Linked in

# **Travis CI**







zendesk







#### MAIN CONCEPTS

- Routing
- Templates
- Components
- Controllers
- Models
- Store

#### ROUTING

```
app/router.js

1 Router.map(function() {
2    this.route('about');
3    this.route('favorites', { path: '/favs' });
4 });
```

```
1 {{#link-to "index"}}<img class="logo">{{/link-to}}
2
3 <nav>
4 {{#link-to "about"}}About{{/link-to}}
5 {{#link-to "favorites"}}Favorites{{/link-to}}
6 </nav>
```

#### NESTED ROUTES

```
app/router.js

1 Router.map(function() {
2    this.route('posts', function() {
3        this.route('favorites');
4    });
5 });
```

```
app/router.js

1 Router.map(function(){
2    this.route('index', { path: '/' });
3    this.route('posts', function() {
4        this.route('index', { path: '/' });
5        this.route('favorites');
6    });
7 });
```

# DYNAMIC SEGMENTS AND ROUTES

```
app/router.js

1 Router.map(function() {
2    this.route('photo', { path: '/photos/:photo_id' });
3   });
```

```
app/routes/photo.js

1 import Ember from 'ember';
2
3 export default Ember.Route.extend({
4    model(params) {
5       return this.get('store').findRecord('photo', params.photo_id);
6    }
7 });
```

```
app/router.js
                                                                      JS
  Router.map(function() {
   this.route('photos');
 });
app/routes/photos.js
                                                                      JS
  import Ember from 'ember';
3
  export default Ember.Route.extend({
    model() {
4
       return this.get('store').findAll('photo');
5
6
  });
```

```
app/templates/photos.hbs

1 <h1>Photos</h1>
2 {{#each model as |photo|}}

3 
4 {{#link-to 'photo' photo.id}}

5 <img src="{{photo.thumbnailUrl}}" alt="{{photo.title}}" />

6 {{/link-to}}

7 
8 {{/each}}
```

#### SEVERAL MODELS IN ROUTE

```
app/routes/songs.js

1  import Ember from 'ember';
2  import RSVP from 'rsvp';
3
4  export default Ember.Route.extend({
5  model() {
6    return RSVP.hash({
7     songs: this.get('store').findAll('song'),
8     albums: this.get('store').findAll('album')
9    });
10  }
11 });
```

```
app/templates/songs.hbs
                                                        HBS
   <h1>Playlist</h1>
  {{#each model.songs as |song|}}
      {{song.name}} by {{song.artist}}
    {{/each}}
  <h1>Albums</h1>
10
11 
    {{#each model.albums as |album|}}
     {{album.title}} by {{album.artist}}
13
    {{/each}}
15
```

#### ROUTES CALLBACKS

```
app/routes/posts.js

1 import Ember from 'ember';
2
3 export default Ember.Route.extend({
4   afterModel(model, transition) {
5    if (model.get('length') === 1) {
6     this.transitionTo('post', model.get('firstObject'));
7   }
8   }
9 });
```

```
app/routes/index.js

1 import Ember from 'ember';
2
3 export default Ember.Route.extend({
4 beforeModel(/* transition */) {
5 this.transitionTo('posts'); // Implicitly aborts the on-going for a part of the content of the conte
```

#### ROUTING SUBSTATES

Loading...

```
app/routes/slow-model.js

1 import Ember from 'ember';
2
3 export default Ember.Route.extend({
4   model() {
5     return this.get('store').findAll('slow-model');
6   }
7 });
```

app/templates/loading.hbs (app/templates/photos/loading.hbs) or app/templates/error.hbs (app/templates/photo/error.hbs)

#### TEMPLATES HANDLEBARS

```
<div>
                                                     HBS
  {{if isFast "zoooom" "putt-putt-putt"}}
  </div>
 2 </div>
  {{#if person}}
                                                      HBS
   Welcome back, <b>{{person.firstName}} {{person.lastName}}</b>!
  {{/if}}
  {{#if person}}
                                                      HBS
   Welcome back, <b>{{person.firstName}} {{person.lastName}}</b>!
  {{else}}
   Please log in.
  {{/if}}
```

#### TEMPLATES: HELPERS

```
/app/templates/components/store-categories.hbs
                                                           HBS
   <l
     {{#each-in categories as |category products|}}
      {{category}}
        <0l>
5
          {{#each products as |product|}}
6
            {{product}}
          {{/each}}
8
        {{/each-in}}
10
```

#### TEMPLATES: HELPERS

```
app/helpers/sum.js

1 import Ember from 'ember';
2
3 export function sum(params) {
4   return params.reduce((a, b) => {
5    return a + b;
6   });
7 };
8
9 export default Ember.Helper.helper(sum);
```

```
1 Total: {{sum 1 2 3}}
```

#### COMPONENTS

1 ember generate component my-component-name

```
app/templates/components/blog-post.hbs

1  <article class="blog-post">
2    <h1>{{title}}</h1>
3    {{yield}}
4    Edit title: {{input type="text" value=title}}
5    </article>
```

```
app/templates/index.hbs

1 {{#each model as |post|}}
2 {{#blog-post title=post.title}}
3 {{post.body}}
4 {{/blog-post}}
5 {{/each}}
```

#### COMPONENTS PARAMS

```
app/templates/index.hbs

1 {{#each model as |post|}}

2 {{blog-post title=post.title body=post.body}}

3 {{/each}}

HBS

HBS
```

**HBS** 

app/templates/components/blog-post.hbs

<div class="body">{{body}}</div>

<h1>{{title}}</h1>

#### DYNAMIC COMPONENTS

```
app/templates/index.hbs

1 {{#each model as |post|}}
2 {{!-- either foo-component or bar-component ---}}
3 {{component post.componentName post=post}}
4 {{/each}}
```

#### COMPONENTS LIFECYCLE

On Initial Render

On Re-Render

On Destroy

- init
- didReceiveAttrs
- willRender
- didInsertElement
- didRender

- didUpdateAttrs
- didReceiveAttrs
- willUpdate
- willRender
- didUpdate
- didRender

- willDestroyElement
- willClearRender
- didDestroyElement

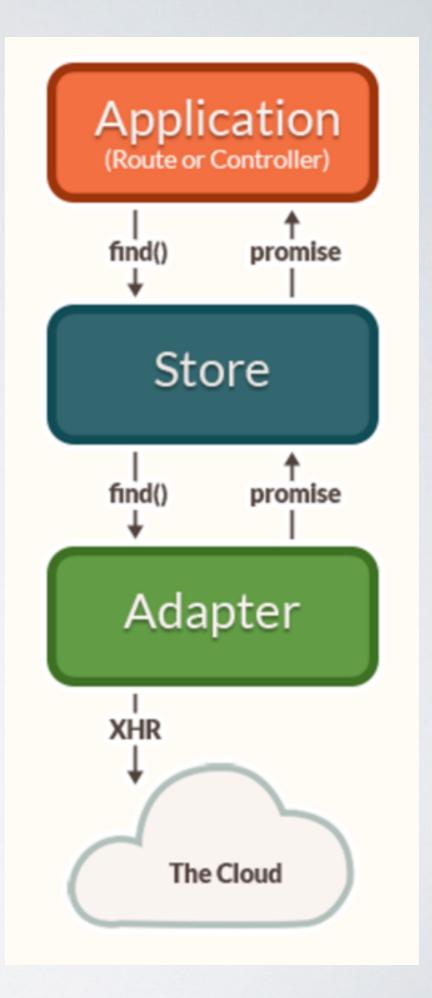
#### MODELS

```
app/models/person.js

1 import DS from 'ember-data';
2
3 export default DS.Model.extend({
4  firstName: DS.attr('string'),
5  birthday: DS.attr('date')
6 });
```

```
app/models/order.js
                                                                      JS
  import DS from 'ember-data';
  export default DS.Model.extend({
   lineItems: DS.hasMany('line-item')
  });
app/models/line-item.js
                                                                     JS
   import DS from 'ember-data';
  export default DS.Model.extend({
   order: DS.belongsTo('order')
  });
```

# THE STORE AND A SINGLE SOURCE OF TRUTH



# JSON

```
1
2
       "posts": {
3
         "id": 5,
         "title": "I'm Running to Reform the W3C's Tag",
4
         "author": "Yehuda Katz",
5
         "comments": [1, 2]
6
7
       "comments": [{
8
9
         "id": 1,
         "author": "User 1",
10
         "message": "First!",
11
12
       }, {
13
         "id": 2,
         "author": "User 2",
14
         "message": "Good Luck!",
15
       }]
16
17
```

# MOVE ON...