Put a little Backbone in your WordPress

http://tunedin.net/ • @roundearth • adam@10up.com
Philosophically, Backbone is an attempt to discover the minimal set of data-structuring (models and collections) and user interface (views and routes) primitives that are generally useful when building web applications with JavaScript.
Why use Backbone?

- Fast, interactive, responsive interfaces
- Organized code, separation of interests
- Lightweight, built to be extended
Should You Use Backbone?
Some Examples

http://backbonejs.org/#examples

Rdio, Hulu, Gawker, Foursquare, Disqus, Khan Academy, Basecamp, Stripe, AirBnB, Pandora, Code School, CloudApp, Trello, Jetpack and on and on…

Backbone in WordPress core…
Backbone in WordPress Core
Backbone Basics

Models
Backbone Basics

Collections
Backbone Basics

Views
More Backbone Basics

- Templating, Events, Routes, History
- Data: bootstrapping & fetching
WordPress and Backbone

- wp_localize_script
- WP JSON REST API
- wp_ajax_(action)
- nonces
- remote api
- transient api
- wp_send_json, wp.template, wp.Backbone.view
Underscore is Awesome

- Collection functions: _.map, _.where, _.pluck, _.filter
- Timing functions: _.memoize, _.throttle, _.debounce, _.once, _.delay, _.defer
- Utility: _.pick, _.extend, _.defaults

http://underscorejs.org/
Underscore Examples

$$\text{gridmodels} = \text{filter( this BackbonePeople.models, \text{function( e )\{\text{self.keyPress( e );}\}, 125 \})}$$
Coding Backbone is fun!
Coding Backbone is fun!

- Great docs and annotated source code
- Easy to get started, small or large projects
- Straightforward and extendable
- Code structure great for collaborating
- Templates, data binding
Challenges & Considerations

➡️ Internationalization & Localization
➡️ Accessibility
➡️ No JavaScript! (nojs)
Building a Simple App

- DEMO 1
- Display a post
- Uses View, Model and wp.template
- Pulls post data from the WP REST API
<!-- Main placeholder div to inject into -->
<div class="backbonedemo"></div>

<!-- Demo post template -->
<script id="tmpl-backbonedemo-post" type="text/html">

<!-- Post -->
<div class="backbonedemo-post-container">
  <div class="backbonedemo-post-title">
    <h2>{{ data.title }}</h2>
  </div>
  <div class="backbonedemo-post-content">
    {{ data.content }}
  </div>
</div>
</script>
/**
 * Bootstrap data for our application using wp_localize_script
 */
$demodata = array(
    'apiurl' => get_json_url(), /* Pass the WP REST API url */
);
wp_localize_script(
    'backbonedemo', /* The script handle */
    'demodata', /* The localized variable name */
    $demodata /* The data to pass */
);
bbdemo.model.PostModel = Backbone.Model.extend( { 

  /**
   * Override the default sync method to call our custom API endpoint.
   */
  sync: function( method, model, options ) {
    console.log( 'bbdemo.model.PostModel::sync' );
    /**
     * Adjust the sync url: set to our api url, Post ID 1.
     */
    options.url = demodata.apiurl + 'posts/1';

    /**
     * Pass back to the default handler.
     */
    return Backbone.sync( method, model, options );
  }
});
Welcome to WordPress. This is your first post. Edit or delete it, then start blogging!

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sapiens autem semper beatus est et est aliquando in dolore; An eum discere ea mavis, quae cum plane perdidercici nihil sciat? Duo Reges: constructio interrete. Torquatus, is qui consul cum Cn. Cur id non ita fit? Quod idem cum vestri faciant, non satis magnam tribuent inventorisibis gratiam. Tibi hoc incredibile, quod beatissimum. Et nunc quidem quod eam tueatur, ut de vite potissimum loquar, est id extrinsecus;"
/**
 * The main post view.
 */

bbdemo.view.PostView = wp.Backbone.View.extend( {

/**
 * Set the view class and template.
 */

    className: 'backbonedemo',
    template: wp.template( 'backbonedemo-post' ),

/**
 * Initialize the view.
 */

    initialize: function( model ) {
        var self = this;
        this.model = model;
        this.model.fetch();
    }
});
Initialization

```javascript
/**
 * MAIN APP SETUP
 */

/**
 * Function to initialize the application.
 */
bbdemo.initialize = function() {
    var postmodel = new bbdemo.model.PostModel();
    bbdemo.view.postview = new bbdemo.view.PostView( postmodel );
    $('body').append( bbdemo.view.postview.el );
}

/**
 * Go! Initialize the application one document ready.
 */
$(document).ready( function() {
    bbdemo.initialize();
});
```
Hello world!

Welcome to WordPress. This is your first post. Edit or delete it, then start blogging! Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sapiens autem semper beatus est et est aliquando in dolore; An eum discere ea mavis, quae cum plane perdiderit nihil sciat? Duo Reges: constructio interrete. Torquatus, is qui consul cum Cn. Cur id non ita fit? Quod idem cum vestri faciant, non satis magnam tribuunt inventoribus gratiam. Tibi hoc incredibile, quod beatissimum. Et nunc quidem quod eam tuetur, ut de vite potissimum loquar, est id extrinsecus;
A bit more complex

- DEMO 2
- Display a post list using bootstrapped data, a template and view
- Click post to load post via the WP REST API
- Uses Collection to handle all posts
- Uses same view from Demo1 for detail view
- Uses Router and History to handle routing
<!-- Demo postlistitem template -->
<script id="tmpl-backbonedemo-postlistitem" type="text/html">
   <div class="backbonedemo-postlistitem-container">
      <div class="backbonedemo-postlistitem-title">
         <h3>
            <a href="#post{{ data.ID }}">{{ data.post_title }}</a>
         </h3>
      </div>
   </div>
</script>
$post_data = get_posts(array(
    'posts_per_page' => 10,
    'no_count_posts' => true,
));
$demodata = array(
    'apiurl' => get_json_url(), /* Pass the WP REST API url */
    'posts' => $post_data,
);
pre-**
*   ---------------------------------------------------------
*  COLLECTIONS
*   ---------------------------------------------------------
* /

bbdemo.PostCollection = Backbone.Collection.extend({});
bbdemo.PostCollectionView = wp.Backbone.View.extend( {  
    /**
     * Set the view class and template.
     */
    className: 'backbonedemo',
    template: wp.template( 'backbonedemo-postlistitem' ),

    /**
     * Initialize the view.
     */
    initialize: function( postCollection ) {
        this.postCollection = postCollection;
        this.render();
    },

    /**
```javascript
refresh: function(id) {
    var self = this;

    this.model = new bbdemo.model.PostModel(id);
    this.model.fetch({
        success: function() {
            self.render();
        }
    });
},

* Handle the postRoute callback - show the post.
*/
```
* Function to initialize the application.

```javascript
bbdemo.initialize = function() {
  /**
   * Set up the post collection with the bootstrapped date.
   */
  var postCollection = new bbdemo.PostCollection( demodata.posts ),

  /**
   * Set up the post collection view, passing the collection.
   */
  postCollectionView = new bbdemo.PostCollectionView( postCollection ),

  /**
   * Set up the Router.
   */
  bbRouter = new bbdemo.Router();
```

Result
Add some moar!

- DEMO 3
- Like Demo 2, now with editing
- Writes data back to WordPress via the WP REST API
Make Content Editable
Handle Edit Events

```javascript
contentEditableChange: function() {

    var titleField = this.$el.find('.backbonedemo-post-title h2'),
        contentField = this.$el.find('.backbonedemo-post-content'),
        titleFieldContent = titleField.html(),
        contentFieldContent = contentField.html();

    /**
     * Save the post if anything has changed.
     */
    if (titleFieldContent !== this.titleFieldContent || contentFieldContent !== this.contentFieldContent) {

        /**
         * Update the stored title/content so we know when it's changed.
         */
        this.titleFieldContent = titleFieldContent;
        this.contentFieldContent = contentFieldContent;
```
Hello World

Welcome to WordPress. This is your first post. Edit or delete it, then start blogging!

Great!
More complex

- Backbone driven plugin to display user directory
- Grabs data by scraping WordCamp attendee page
- Data stored in CPT, passed via JSON REST API
- Search feature, grid and detail views

https://github.com/adamsilverstein/backbone-directory
Start Using Backbone

Yum, Pie!

- Simple concepts, easy to start
- Fast and lightweight
- Flexible and extendible
- Used extensively in WordPress core
About

• Saved assembly code onto cassettes
• Good at JavaScript & bug squashing
• Love contributing to WordPress Core
• WP Revisions Component Maintainer
• Senior Web Engineer at 10up

Adam Silverstein

http://tunedin.net/ • @roundearth • adam@10up.com
Resources

Great resources to extend Backbone:

* https://github.com/jashkenas/backbone/wiki/Extensions,-Plugins,-Resources

Some highlights, lots of data binding and:

* http://exosjs.com/
* https://github.com/jhudson8/react-backbone
* https://github.com/jeromegn/Backbone.localStorage
* https://github.com/andreypopp/backbone.projections
* https://github.com/adamrneary/backbone-memoize
* https://github.com/gsamokovarov/grunt-init-backbone-plugin
Resources

- Backbone - http://backbonejs.org/
- Underscore - http://underscorejs.org/
- Code School Backbone course - https://www.codeschool.com/paths/javascript#backbone-js
- Developing Backbone Applications - https://www.codeschool.com/paths/javascript#backbone-js
- TLC Transients - github.com/markjaquith/WP-TLC-Transients
- Demo Code: https://github.com/adamsilverstein/backbone-demo