



# Put a little Backbone in your WordPress







# Backbone is Awesome

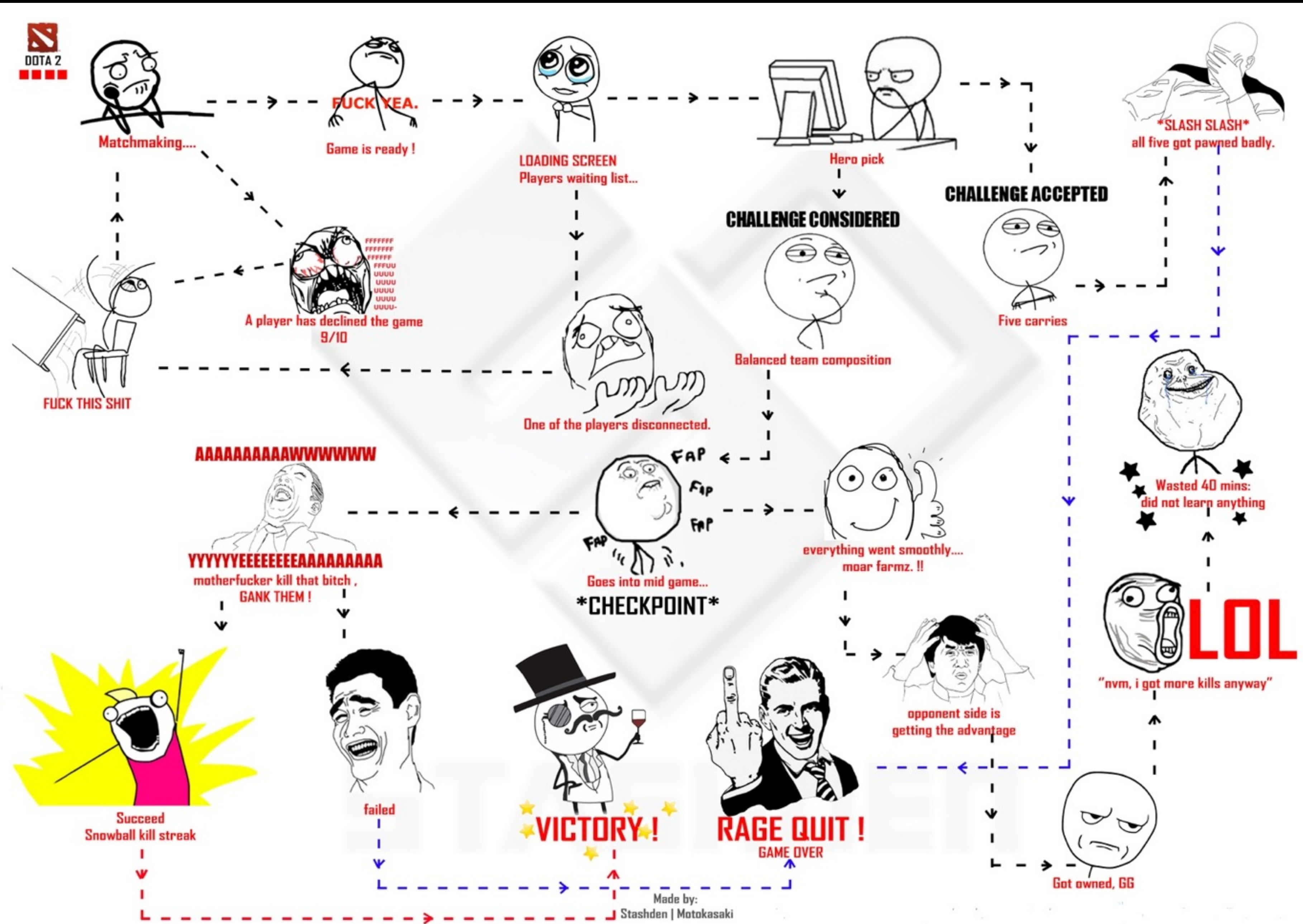
*Philosophically, Backbone is an attempt to discover the minimal set of data-structuring (models and collections) and user interface (views and routes) primitives that are generally useful when building web applications with JavaScript*

# 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

```
gridmodels = filter( this backbonePeople models  
$( document ).on( 'keyup', _.throttle( function( e ) {  
  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

# Template

```
<!-- 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

```
/**
 * 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 */
);
```



# Post Model

```
/**
 * =====
 * MODELS
 * =====
 */
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 );
  }
} );
```



# JSON REST API

```
{
  ID: 1,
  title: "Hello world!",
  status: "publish",
  type: "post",
  author: {...},
  content: "Welcome to WordPress. This is your first post. Edit or delete it, then start blogging!\r\n\r\nLorem 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 perdidiceriti 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;",
  parent: null,
  link: http://wp.localhost/?p=1,
  date: "2015-03-14T19:49:25",
  modified: "2015-03-15T20:16:00",
  format: "standard",
  slug: "hello-world",
  guid: http://wp.localhost/?p=1,
  excerpt: "<p>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 perdidiceriti nihil sciat? Duo Reges: constructio interrete. Torquatus, is qui consul cum Cn. Cur [ &hellip; ]</p>\n",
  menu_order: 0,
  comment_status: "open",
  ping_status: "open",
  sticky: false,
  date_tz: "UTC",
  date_gmt: "2015-03-14T19:49:25",
  modified_tz: "UTC",
  modified_gmt: "2015-03-15T20:16:00",
  meta: {...},
  featured_image: null,
  terms: {...}
}
```



# Post View

```
/**
 * 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;
  }
});
```



# Initialization

```
/**
 * =====
 * MAIN APP SETUP
 * =====
 */

/**
 * Funciton 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();
});
```

# Result

## **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 perdidiceriti 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;



# Result

```
<!-- 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>
  </script>
```

# 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



# Template

```
<!-- 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>
```

# Bootstrap Data

```
$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,  
);
```



# Post Collection

```
/**
 * =====
 * COLLECTIONS
 * =====
 */
bbdemo.PostCollection = Backbone.Collection.extend( {} );
```

# Post Collection View

```
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();  
  },  
  
  /**
```



# Router

```
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.  
*/
```

# Initialization

```
* Function to initialize the application.  
*/  
bbdemo.initialize = function() {  
    /**  
     * Set up the post collection with the bootstrapped data.  
     */  
    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

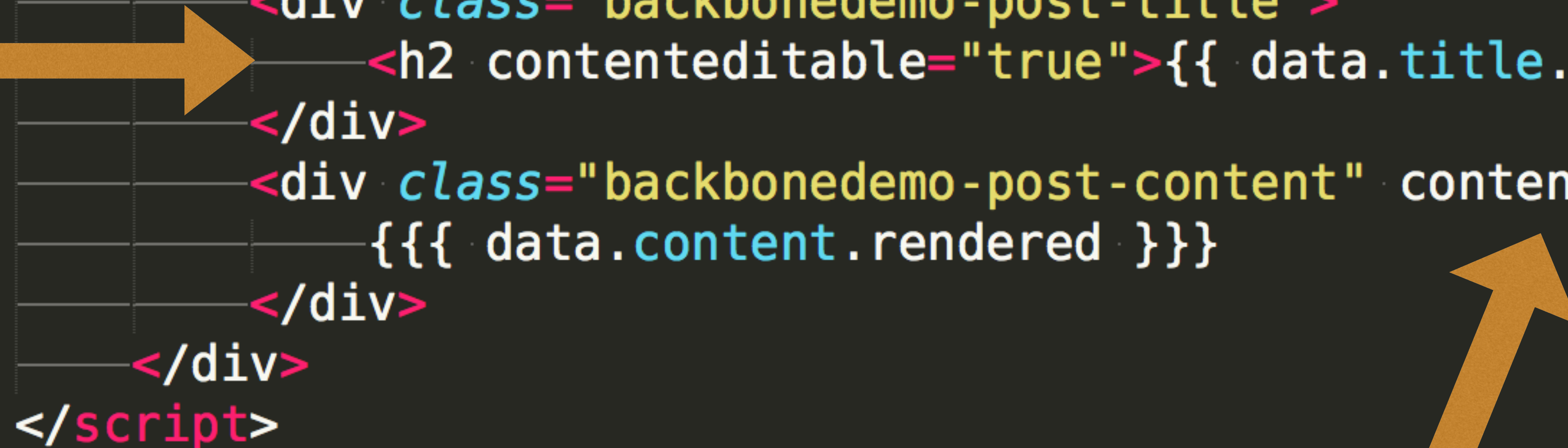
➡ Like Demo 2, now with editing

➡ Writes data back to WordPress via the WP REST API



# Make Content Editable

```
<!-- Demo post template -->
<script id="tmpl-backbonedemo-post" type="text/html">
  <!-- Post -->
  <div class="backbonedemo-post-container" >
    <div class="backbonedemo-post-title">
      <h2 contenteditable="true">{{ data.title.rendered }}</h2>
    </div>
    <div class="backbonedemo-post-content" contenteditable="true">
      {{{ data.content.rendered }}}
    </div>
  </div>
</script>
```

Two orange arrows are present. One arrow on the left points horizontally to the `contenteditable="true"` attribute on the `<h2>` tag. The other arrow on the bottom right points diagonally upwards to the `contenteditable="true"` attribute on the `<div class="backbonedemo-post-content">` tag.

# Handle Edit Events

```
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 its changed.
     */
    this.titleFieldContent = titleFieldContent;
    this.contentFieldContent = contentFieldContent;
  }
}
```



## Hello World

Post 7

Post 6

Post 5

Post 4

Post 3

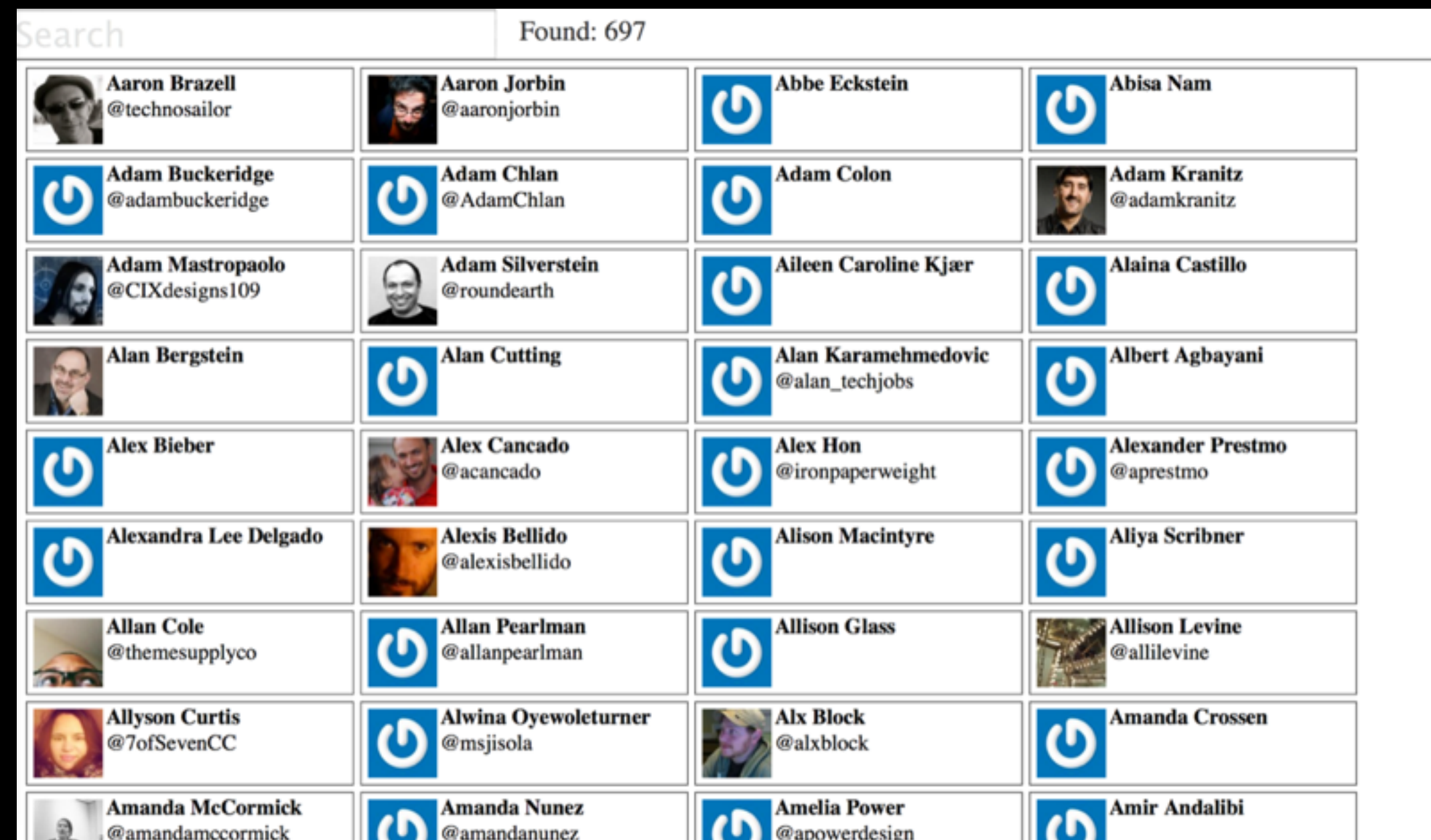
## Hello World

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

# 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>
- JSON REST API - <https://wordpress.org/plugins/json-rest-api/> & <https://github.com/WP-API/WP-API>
- TLC Transients - [github.com/markjaquith/WP-TLC-Transients](https://github.com/markjaquith/WP-TLC-Transients)
- wp.Backbone.View/Subview - <http://wordpress.tv/2014/11/03/mark-jaquith-backbone-views-in-wordpress/>
- Demo Code: <https://github.com/adamsilverstein/backbone-demo>

