

Backbone.js

An Introduction...

Wait - Who *is* this guy?

- ✦ Jim Cowart - Sr. Architect & Developer @ appendTo
- ✦ Street Cred - <http://github.com/ifandelse>
- ✦ Fake Street Cred - <http://freshbrewedcode.com/jimcowart>

What is Backbone.js?

- ✦ Client-side framework
 - ✦ models
 - ✦ collections
 - ✦ views
 - ✦ router
 - ✦ RESTful-ish persistence

Benefits

- ✦ Promotes good SoC
- ✦ You retain control over what matters
- ✦ Opinionated “enough” - but very extensible
- ✦ Plays well with jQuery, require.js and more

“When working on a web application that involves a lot of JavaScript, one of the first things you learn is to stop tying your data to the DOM. It's all too easy to create JavaScript applications that end up as tangled piles of jQuery selectors and callbacks, all trying frantically to keep data in sync between the HTML UI, your JavaScript logic, and the database on your server. For rich client-side applications, a more structured approach is often helpful.”

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

That being said...

- Good Fit For - assisting server-side web frameworks (PHP, Rails, ASP.NET, etc.) in producing rich client functionality
- *Great* Fit For
 - any web application taking advantage of client-side routing (hash/push state)
 - web applications that take advantage of modular + optimized assets (templates, etc.)
 - teams tired of typical DOM/jQuery spaghetti + 'mother-of-all-JavaScript' includes
- Not really a great fit for static/brochure style sites

And now for some `<code />`

- ✦ Let's look at a view (<http://jsfiddle.net/ifandelse/5WE9h/>):
 - ✦ how to create
 - ✦ how DOM elements are constructed/targeted
 - ✦ view selector
 - ✦ event delegation

more `<code />`

- ✦ Let's look at a model (<http://jsfiddle.net/ifandelse/Y8atH/>):
 - ✦ how to create
 - ✦ setting defaults & initializing
 - ✦ events
 - ✦ validation

Ok that's great and all..

...but can we see a sample app now?

Ok that's great and all..

...but can we see a sample app now?

“Recipe” sample app:

<https://github.com/ifandelse/Backbone-Introduction>

The good & the bad of grokking Backbone.js

The Good

- ✦ Encapsulated views are addictive.
- ✦ Event delegation conventions
- ✦ Model-to-View eventing is great start.
- ✦ Route matching + param support
- ✦ Built in Backbone.sync is nearly painless

The Gotchas

- ✦ Heavy/complicated model-to-view eventing gets verbose
- ✦ Backbone.sync overrides can be tricky/break idioms
- ✦ No higher level View Management
- ✦ Model Validation validates entire model, not just the attributes being set

Roll Your Own Or Download it

- ✦ (or both!)
- ✦ “Meta” frameworks exist to address some of the gotchas:
 - ✦ Backbone.Marionette
 - ✦ <https://github.com/derickbailey/backbone.marionette>
 - ✦ Backbone Boilerplate
 - ✦ <https://github.com/tbranyen/backbone-boilerplate>

Backbone Resources

- <http://documentcloud.github.com/backbone/>
 - Check out the annotated source!
 - Lots of example applications to review
 - <https://github.com/documentcloud/backbone/wiki>
- <http://backbonetraining.net/resources>
- <http://backbonefu.com/>
- <http://backbonetutorials.com/>
- <http://www.quora.com/What-are-some-good-resources-for-Backbone-js>