

A Subtle Benefit Of ES6 Arrow Functions With Backbone/Marionette

July 21, 2015 By [derickbailey](#)

I started a new [Backbone/Marionette](#) application this week, and decided to put Babel in place so I can use ES6 features (if you want an introduction to Babel / ES6 features, check out [my WatchMeCode series on ES6](#)). I honestly wasn't sure what benefit I would get, at first. I've been playing with ES6 for a while now but have not yet found the "AHA!" moment that makes me really want to use it everywhere.

What I have found, however, is a subtle benefit of using arrow functions with Backbone/Marionette, that makes me smile every time.

Managing “this” With Arrow Functions

One of the nice features of arrow functions is the ability to manage “this” in the function being executed. Arrow functions use lexical scoping to ensure “this” is maintained based on the context of the function being called (see this [WatchMeCode episode on arrow functions](#) for more details).

In short, that allows me to take this Backbone/Marionette code:

```
1  var TitleView = Marionette.ItemView.extend(){
2
3    showTitle : function (region){
4      var title = new Title();
5      title.on("entry:add ", function(){
6        this.addEntry();
7      }, this);
8      region.show(title);
9    },
10
11    addEntry : function(){
12      var addEntry = new AddEntryForm();
13      this.region.show(addEntry);
14    }
15
16  });
```

1.js hosted with ❤ by GitHub

[view raw](#)

and turn it in to this:

```
1  var TitleView = Marionette.ItemView.extend(){
2
3    showTitle : function (region){
4      var title = new Title();
5      title.on("entry:add ", () => this.addEntry());
6      region.show(title);
7    },
8
9    addEntry : function(){
10      var addEntry = new AddEntryForm();
```

```
11 } this.region.show(addEntry);  
12  
13  
14 });
```

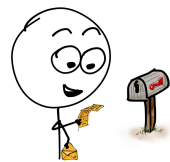
2.js hosted with ❤ by GitHub

[view raw](#)

The difference is small... it's a bit subtle, if you're not looking for it. But it's right there on line 5.

Subtle, And Important

The difference of manually specifying "this" as a parameter to the event handlers having it managed for me through the arrow function is important in my mind. It prevents me from having to manually manage and manipulate the value of "this" in many circumstances. And while this isn't the "AHA!" moment that makes me swear ES6 is the greatest thing to happen to JavaScript, it does provide enough value that it warrants the additional layer of BabelJS transpilation in my mind.



Learn JavaScript's Secrets

Join 5000+ Developers on
Derick Bailey's Mailing List
and get everything you need
to know about JavaScript
sent straight to your inbox!

**SEND ME THE
SECRETS!**

[Tweet](#)

RELATED POST

**The Docker
Management
Cheatsheet**

**What Is RabbitMQ?
What Does It Do
For Me?**

**Callbacks First,
Then Promises**

**Ending the Nested
Tree of Doom with
Chained Promis...**

**Does ES6 Mean
The End Of
Underscore /
Lodash?**

Filed Under: [Arrow Functions](#), [Backbone](#), [ES6](#), [JavaScript](#),
[MarionetteJS](#)



About derickbailey

Derick Bailey is a developer, entrepreneur, author, speaker and technology leader in central Texas (north of Austin). He's been a professional developer since the late 90s, and has been writing code since the late 80s. In his spare time, he gets

called a spamming marketer by people on Twitter, and blurts out all of the stupid / funny things he's ever done in his career on [his email newsletter](#)

DERICK BAILEY AROUND THE WEB

Twitter: [@derickbailey](#)

Google+: [DerickBailey](#)

Screencasts: [WatchMeCode.net](#)

eBook: [Building Backbone Plugins](#)