

AngularJS Directives

DSL For Your HTML!

Who are we?

Who am I?

Lukas Ruebbelke

Senior Software Engineer at WebFilings

Contributed to AngularJS

Co-Author of AngularJS in Action

Who is he?

Brian Ford

Core Contributor to AngularJS

Core Contributor to Yeoman

Co-Author of AngularJS in Action

Wrote Batarang

Loveliest Hair - Univeristy of Michigan 2011



Brian Ford === JoJo

**Inspiration behind Horton Hears a Who Character
True Story Bro!**



Okay so what is AngularJS?

“AngularJS lets you extend HTML vocabulary for your application.”

JavaScript Framework

'Model View Whatever'

For the sake of conversation we refer W as ViewModel

ViewModel increases surface area for testable code

Two Way Databinding

DOM is NOT the model - Keep state out of your DOM

DOM should only visually reflect the state of your app

Creates a nice separation between declarative markup and imperative functions

More productive way to write a web app

Plain JavaScript

You are not locking your JavaScript into special container classes

No API to learn

Templating

Don't try to solve HTML by abstracting it away

Embrace it and extend it

No redraw or refresh

Dependency Injection

Have what you need delivered to you

Great for mocks

Encourages testable code

Testable

Written from the ground up to be entirely testable

Any assertion library will do

Full E2E with Karma

Embeddable

Plays nice with others

Not an all or nothing proposition

What is a DSL?

A Domain Specific Language (DSL) is a programming language designed specifically to express solutions to problems in a specific domain.

But Why?

*“ People find DSLs valuable because a well-designed DSL can be much easier to program with than a traditional library. This improves programmer productivity, which is always valuable. **In particular it may also improve communication with domain experts, which is an important tool for tackling one of the hardest problems in software development.** ”*

Martin Fowler

Fixed Languages Are Broken

“ [Without the ability to reprogram the language] programmers resort to repetitive, error-prone workarounds. Literally millions of lines of code have been written to work around missing features in programming languages. ”

Stuart Halloway

**HTML is a fixed language... OR IS
IT!?!?**

Anatomy of a Directive

What is a Directive? What does it do? What are its parts?

What is the difference between a Directive and a Recommendation?

What is the difference between a Directive and a Regulation?

What is the difference between a Directive and a Decision?

What is the difference between a Directive and a Treaty?

What is the difference between a Directive and a Law?

What is the difference between a Directive and a Statute?

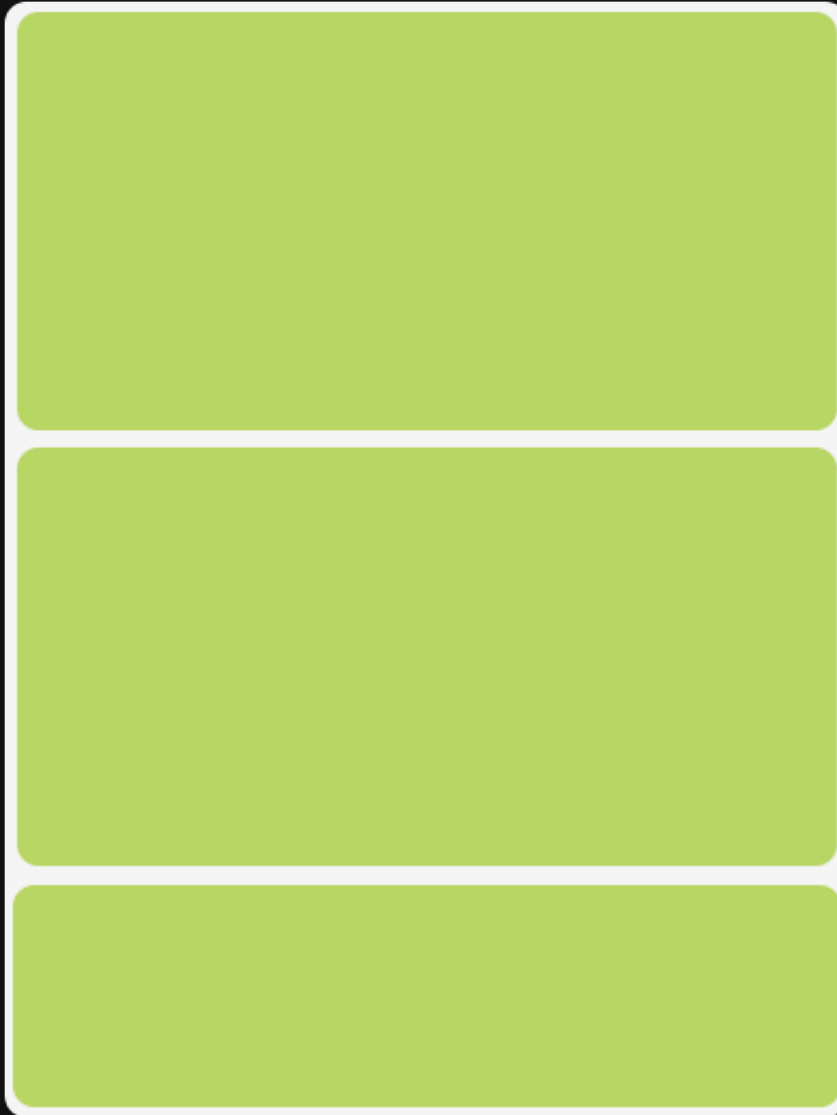
What is the difference between a Directive and a Decree?

What is the difference between a Directive and an Order?

What is the difference between a Directive and a Circular?

What is the difference between a Directive and a Memorandum?

What is the difference between a Directive and a Resolution?



LINK

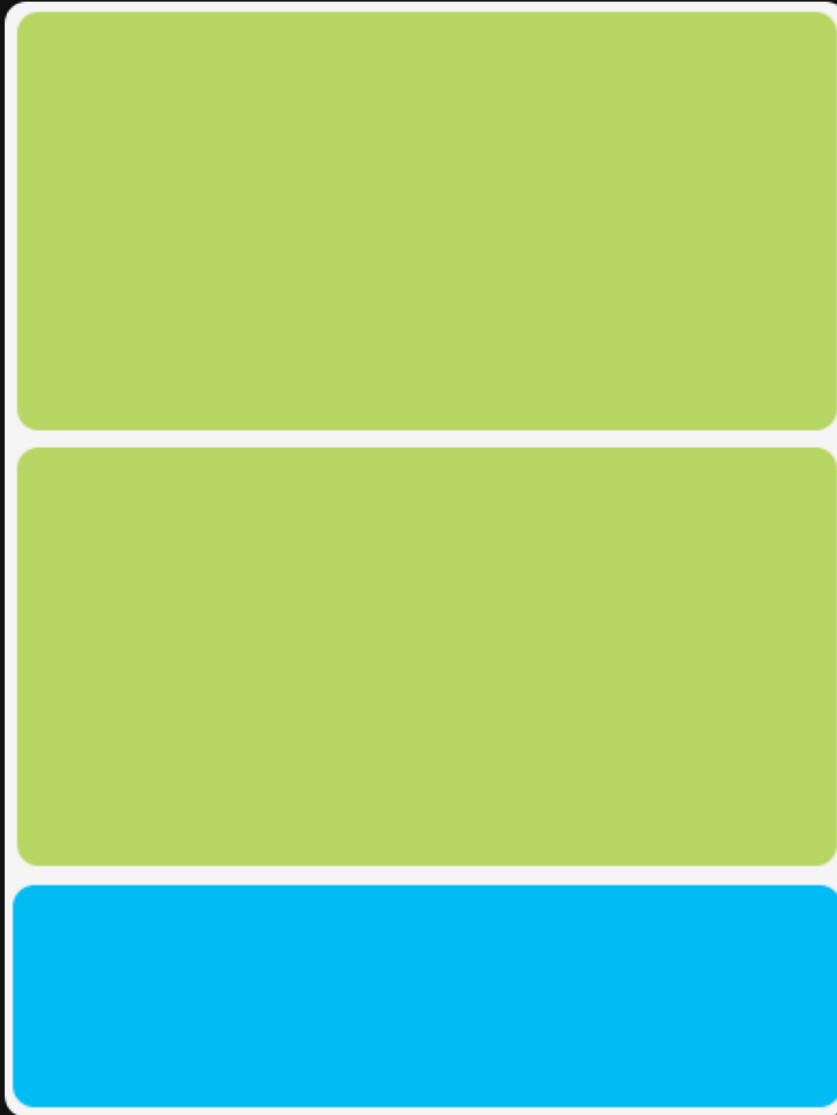
CONTROLLER

DDO

```
myModule.directive('sampleDirective', function(){
  var linker = function(scope, element, attrs) {
    // FASHION GOES HERE
  };

  var controller = function(scope, element, attrs) {
    // FUNCTIONALITY GOES HERE
  };

  // DIRECTIVE DEFINITION OBJECT (DDO)
  return {
    restrict: 'A',
    link: linker
    controller: controller
  }
})
```



LINK

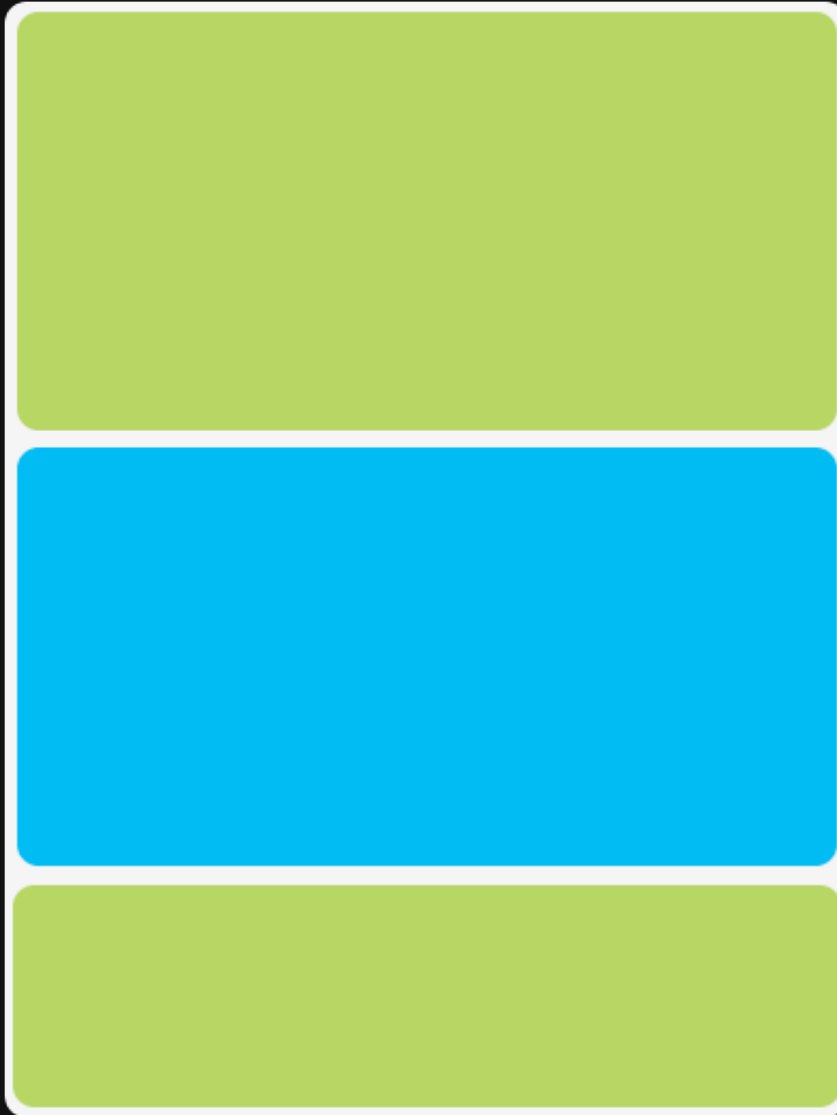
CONTROLLER

DDO

```
myModule.directive('sampleDirective', function(){
  var linker = function(scope, element, attrs) {
    // FASHION GOES HERE
  };

  var controller = function(scope, element, attrs) {
    // FUNCTIONALITY GOES HERE
  };

  // DIRECTIVE DEFINITION OBJECT (DDO)
  return {
    restrict: 'A',
    link: linker
    controller: controller
  }
})
```



LINK

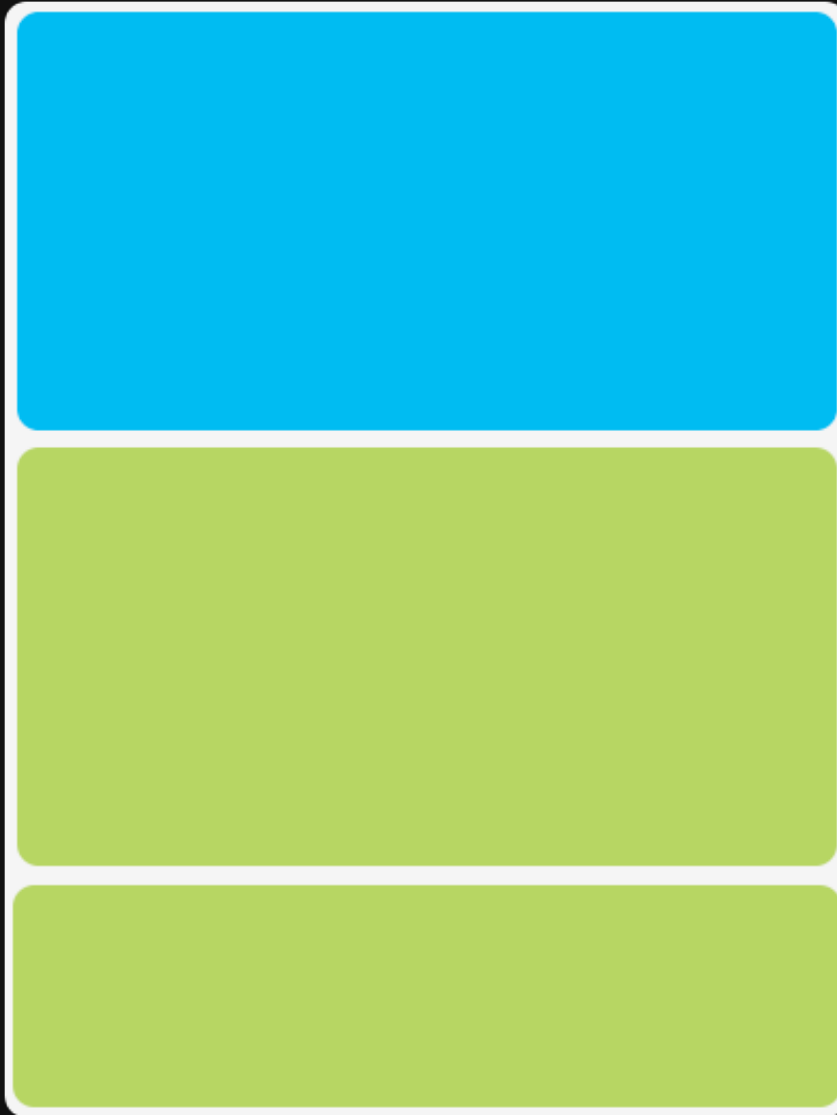
CONTROLLER

DDO

```
myModule.directive('sampleDirective', function(){
  var linker = function(scope, element, attrs) {
    // FASHION GOES HERE
  };

  var controller = function(scope, element, attrs) {
    // FUNCTIONALITY GOES HERE
  };

  // DIRECTIVE DEFINITION OBJECT (DDO)
  return {
    restrict: 'A',
    link: linker
    controller: controller
  }
})
```

LINK

CONTROLLER

DDO

Restrict

E is for ELEMENT

A is for ATTRIBUTE



C is for COOKIE!

C is for CLASS

Isolated Scope

Why?

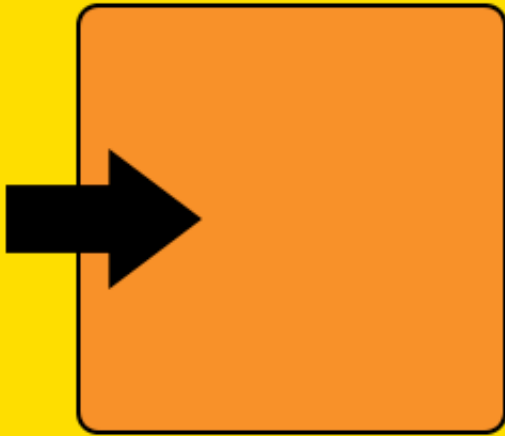
Explicit Interface

Reusable

Mitigates Side Effects


```
myModule.directive('myComponent', function () {  
  return {  
    restrict: 'E',  
    scope: {  
      attributeFoo: '@',  
      bindingFoo: '=',  
      expressionFoo: '&  
    }  
  };  
})
```

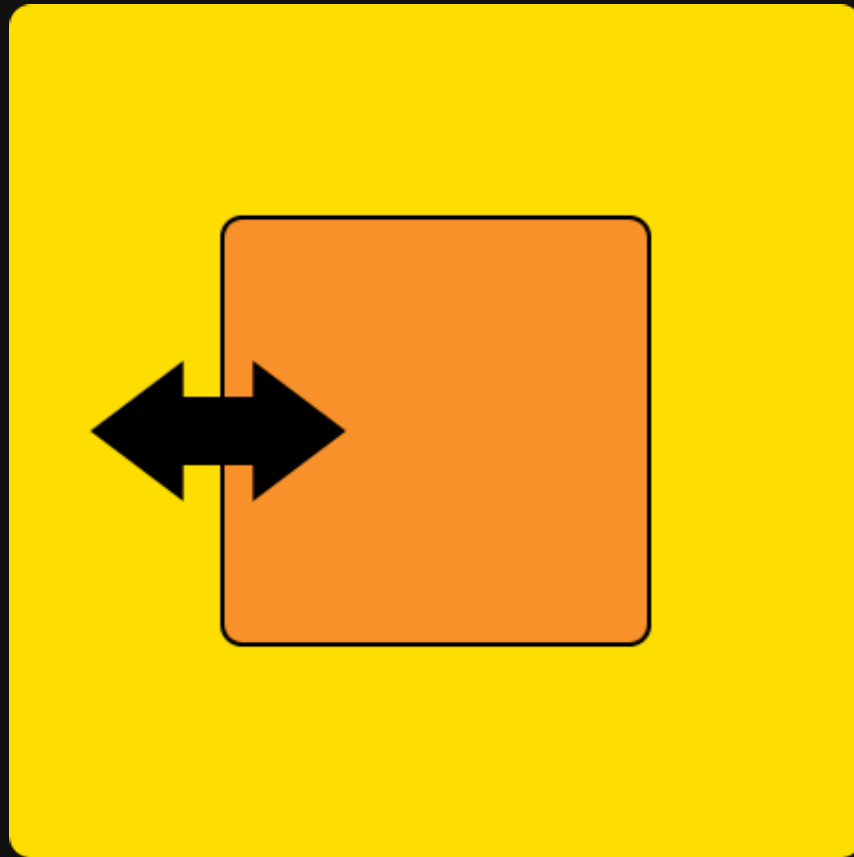
Attribute



**“I’ll let you know
when this has
changed”**

“Cool.”

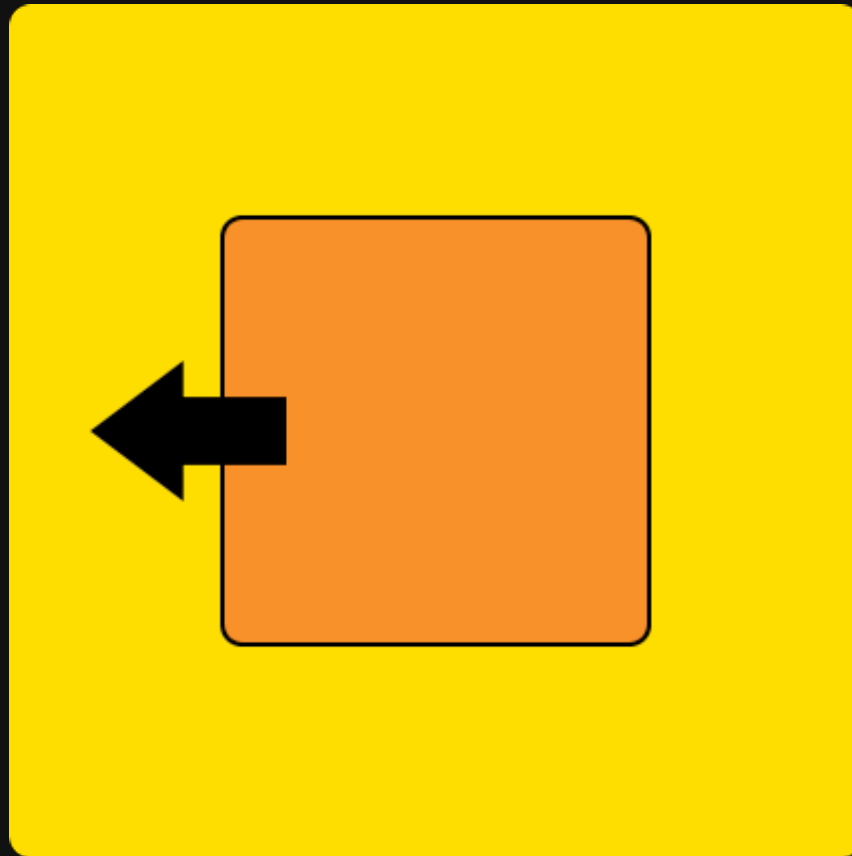
Binding



**“Let’s keep
each other in
the loop.”**

“Bromance.”

Expression



**“Let me know
if you need me
to do something.”**

**“Go get me a
juice box.”**



Stop! Demo Time!

Questions?

Holla At Yer Boy!!

BLOG: <http://onehungrymind.com/>

GITHUB: <https://github.com/simpulton>

TWITTER: @simpulton

Resources

<http://angularjs.org/>

<http://onehungrymind.com/>

<http://briantford.com/>

<http://egghead.io/>

AngularJS in Action



Chapters 3 and 4 In Review

Angello Posted

Coupon Code **12mp25** for 42% Off

Thanks!

