



angularJS Directives

Shared/Isolate Scope and the link() Function

Understanding Shared and Isolate Scope

@ Local Scope Property

= Local Scope Property

& Local Scope Property

Understanding Shared and Isolate Scope



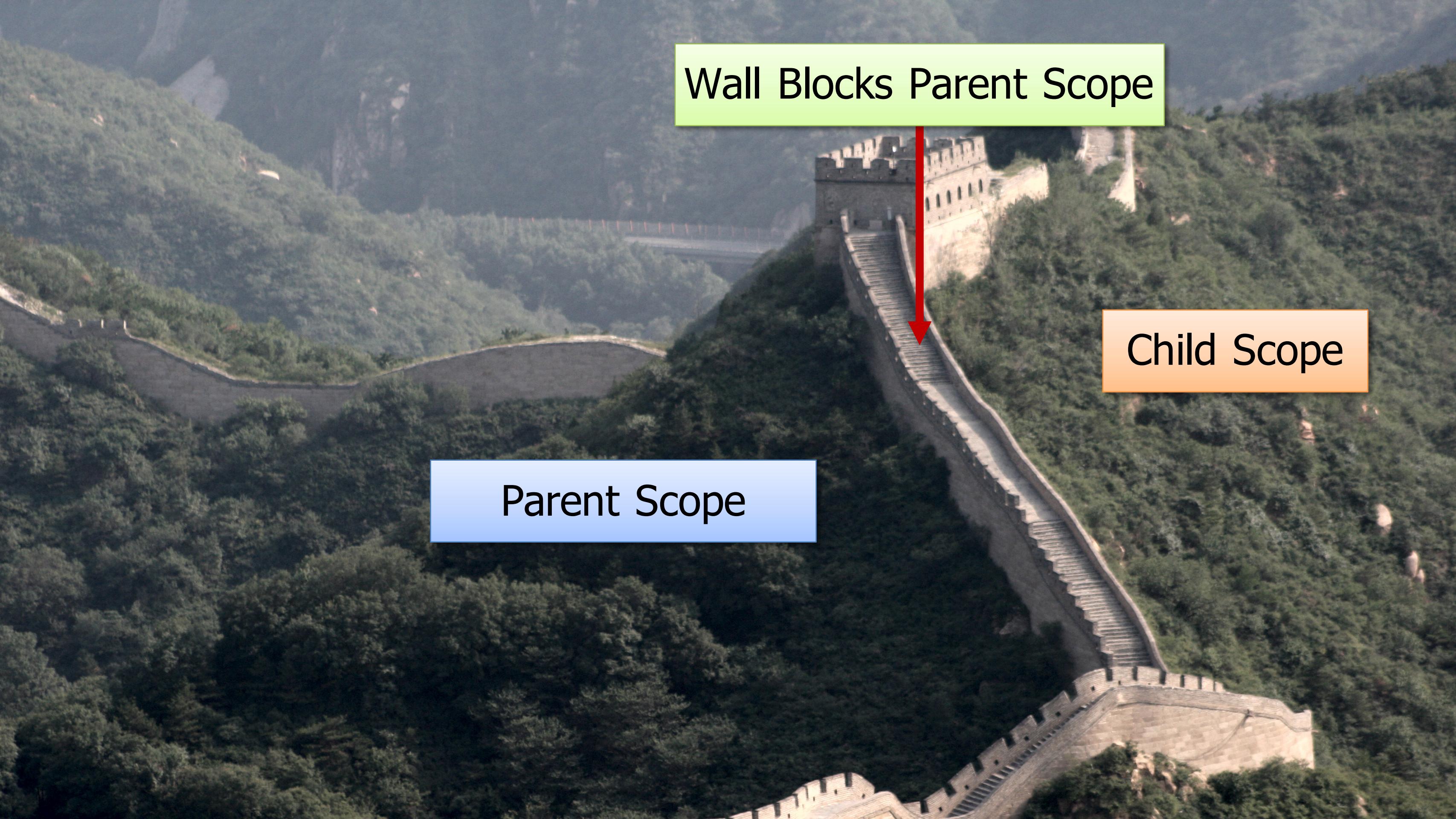
Parent Scope

Shared Scope

Child Scope

A photograph of a person sitting on a light-colored couch, appearing distressed. They are wearing a white long-sleeved shirt, dark jeans, and green and white striped socks. Their head is buried in their hands, and they are looking down. The background shows a tiled floor.

Isolate Scope



Wall Blocks Parent Scope

Parent Scope

Child Scope

Shared Scope

Controller

```
$scope.customers=[];
```

Shared

```
$scope.customers=[];
```

Directive

Depends on parent scope

Isolate Scope

Controller

```
$scope.customers=[];
```

Not Shared

```
$scope.customers=[];
```

```
$scope.isolated = true;
```

Directive

Isolated from parent scope

Shared Scope Directive Example

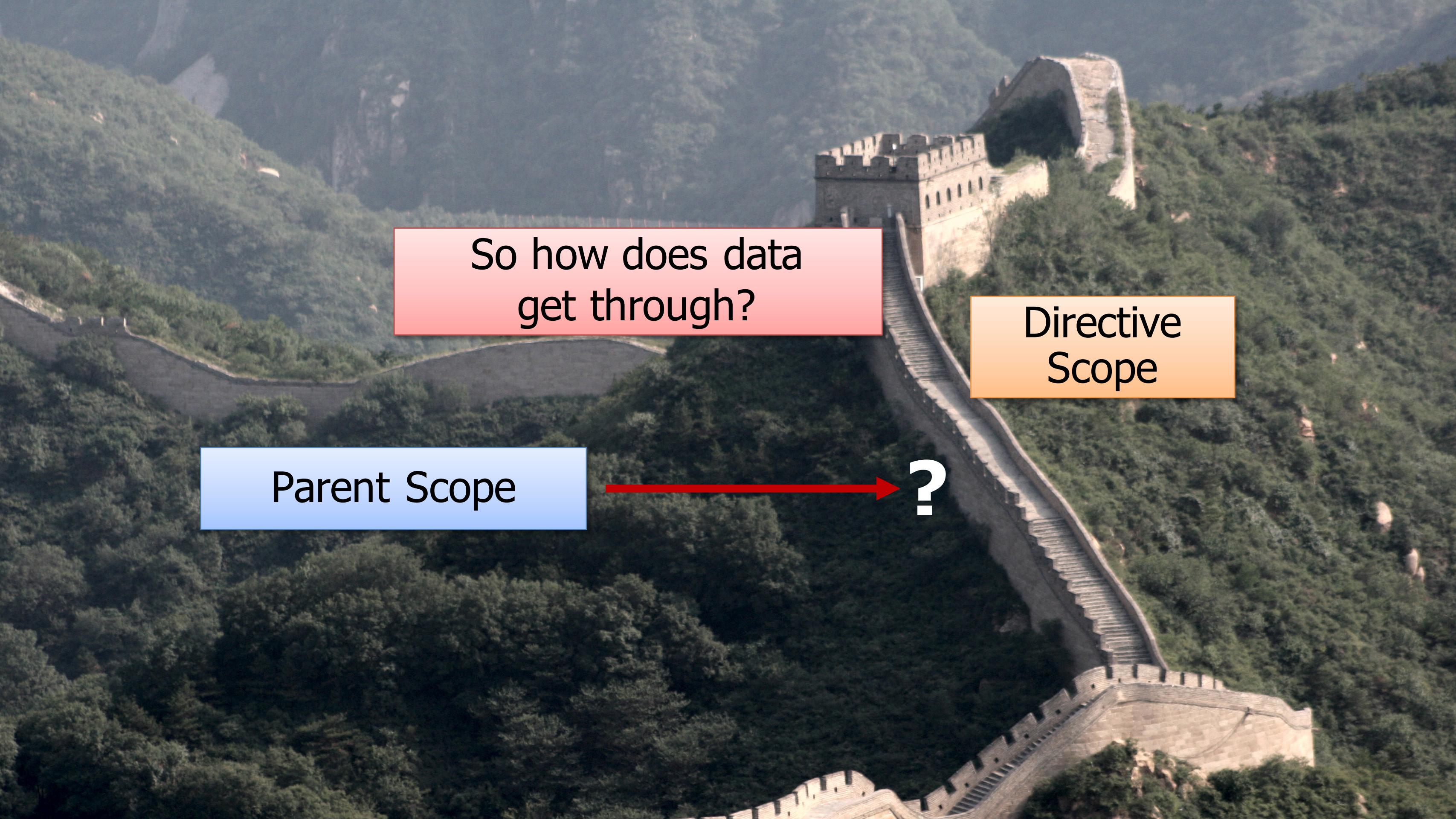
```
var app = angular.module('myModule', []);  
  
app.controller('Controller', ['$scope', function ($scope) {  
    $scope.customer = {  
        name: 'David',  
        street: '1234 Anywhere St.'  
    };  
}]);  
  
app.directive('sharedScope', function () {  
    return {  
        template: 'Name: {{customer.name}} Street: {{customer.street}}'  
    };  
});  
  
Scope is inherited  
<div shared-scope></div>
```

Isolating Scope in Directives

```
angular.module('myModule')
  .directive('isolateScope', function () {
    return {
      scope: {}, //isolate scope
      template: 'Name: {{customer.name}} Street: ' +
                 '{{customer.street}}'
    };
});
```

<div isolate-scope></div>

**No data will
display!**



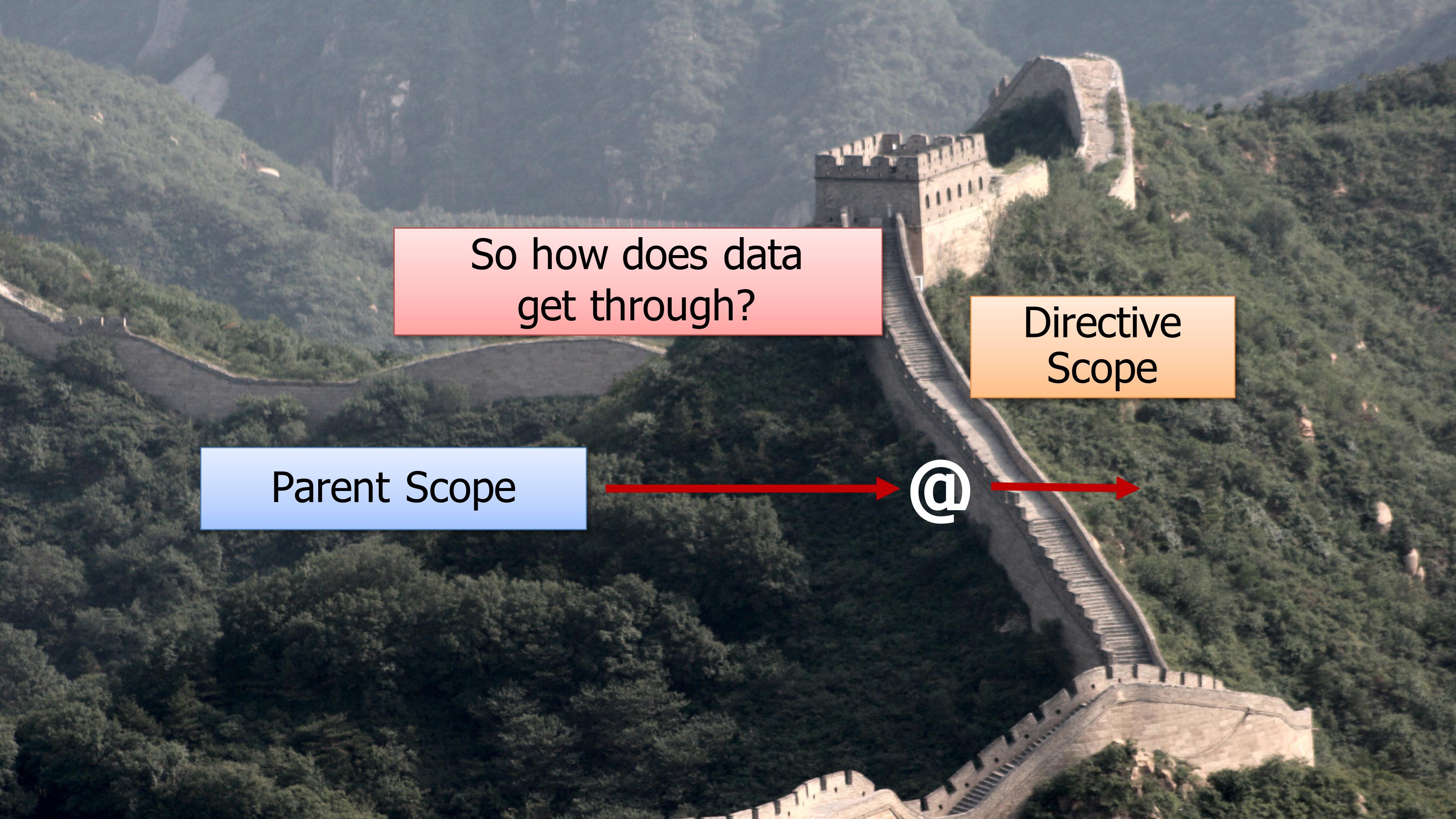
So how does data
get through?

Directive
Scope

Parent Scope



@ Local Scope Property



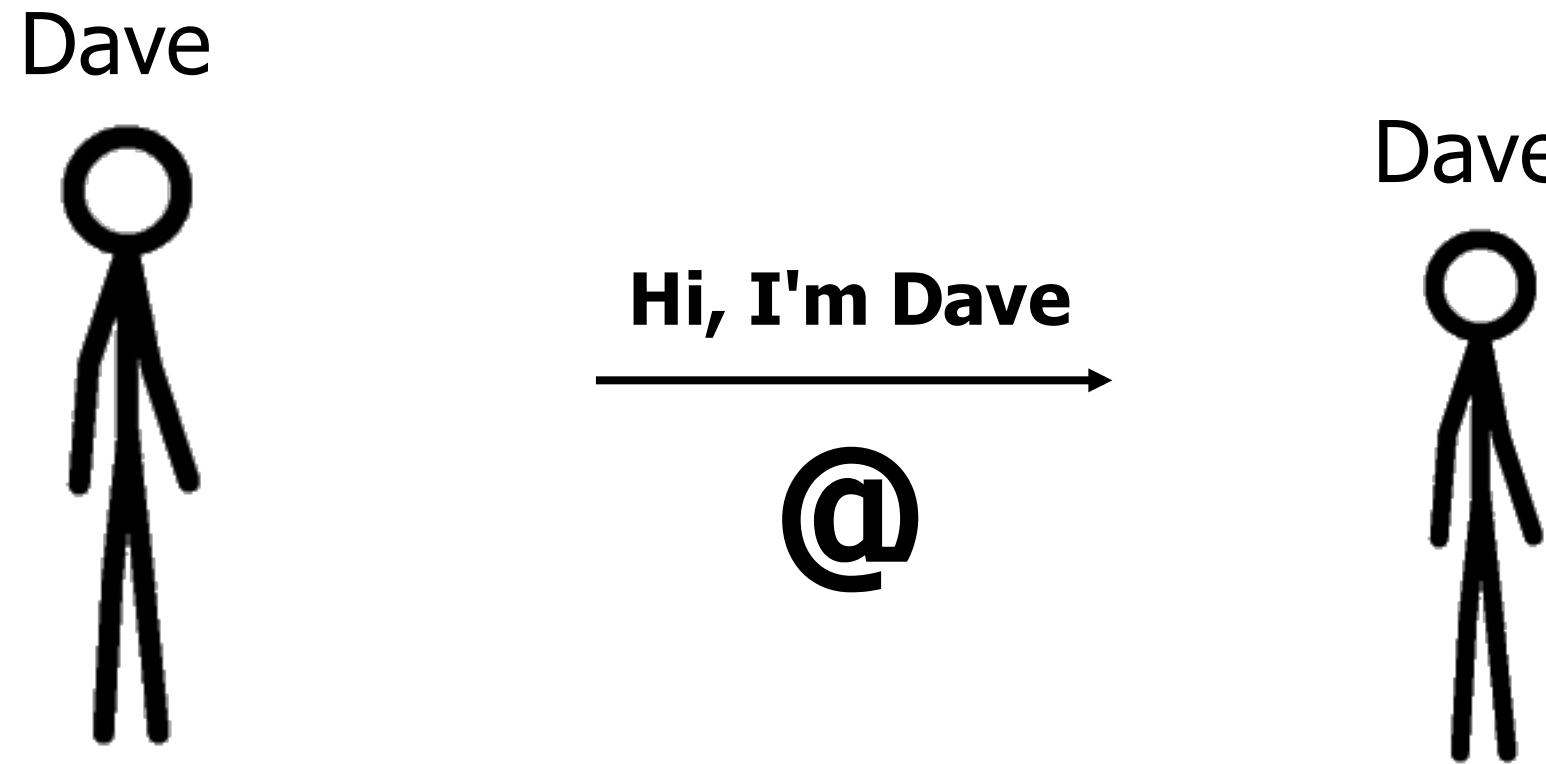
So how does data
get through?

Directive
Scope

Parent Scope

@

@ Local Scope Property



String with one-way binding

@ local scope property

```
$scope.first='Dave';
```

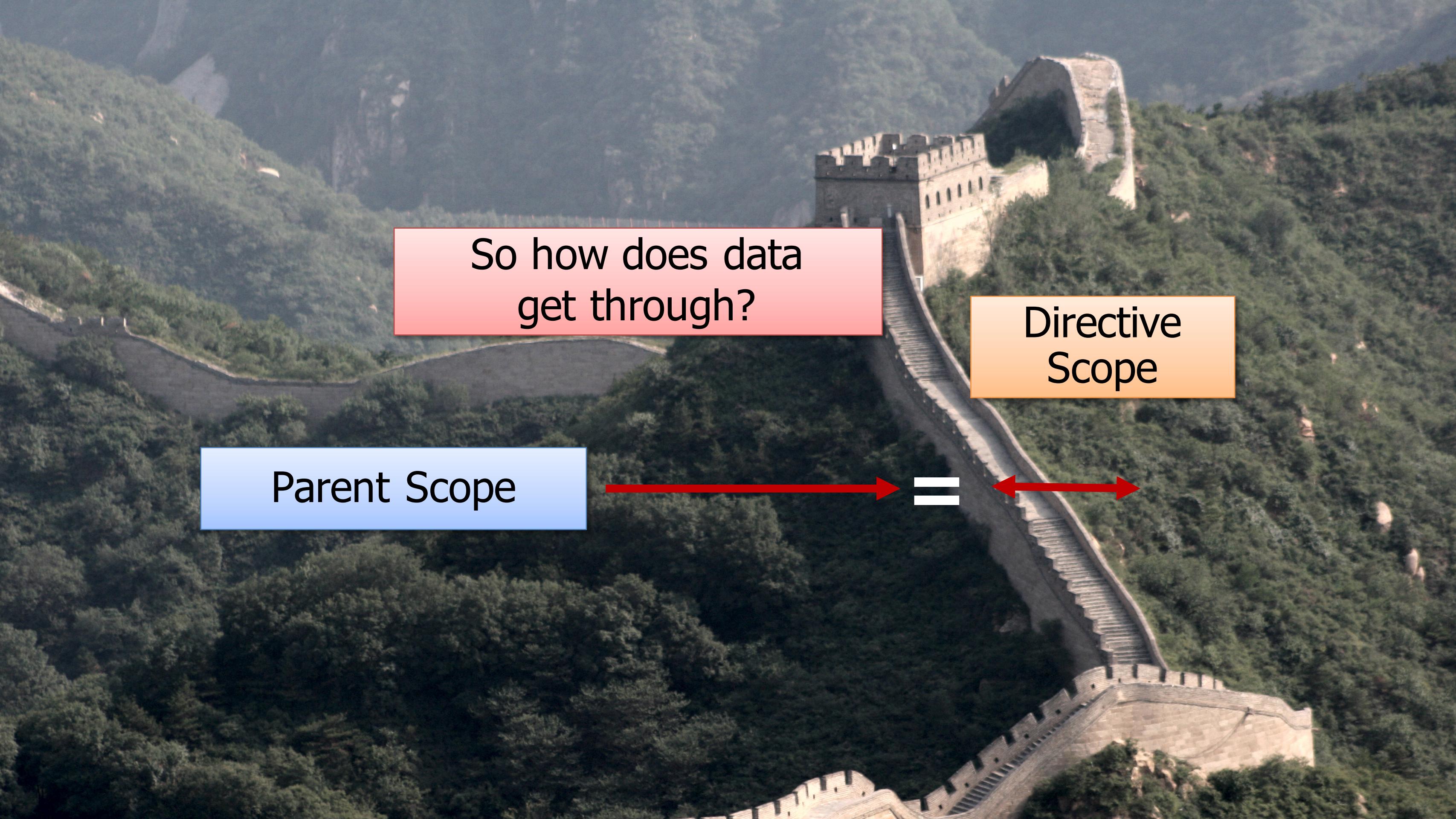
```
<my-directive name="{{first}}"/>
```

scope: { name: '@' }

Directive

Directive can access a string value

= Local Scope Property



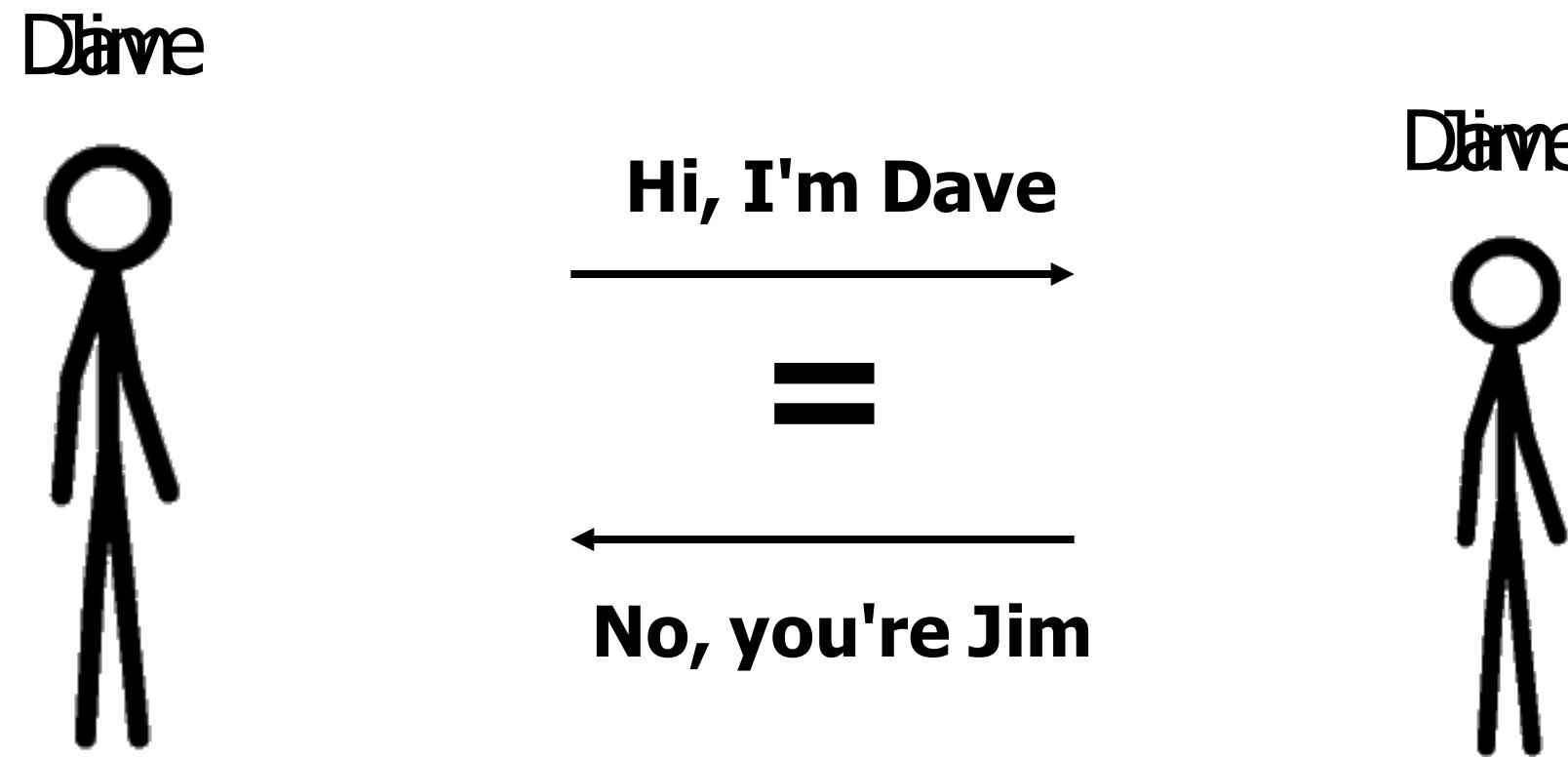
So how does data
get through?

Directive
Scope

Parent Scope



= Local Scope Property



Bi-directional Binding

= local scope property

```
$scope.person={name:'Dave'};
```

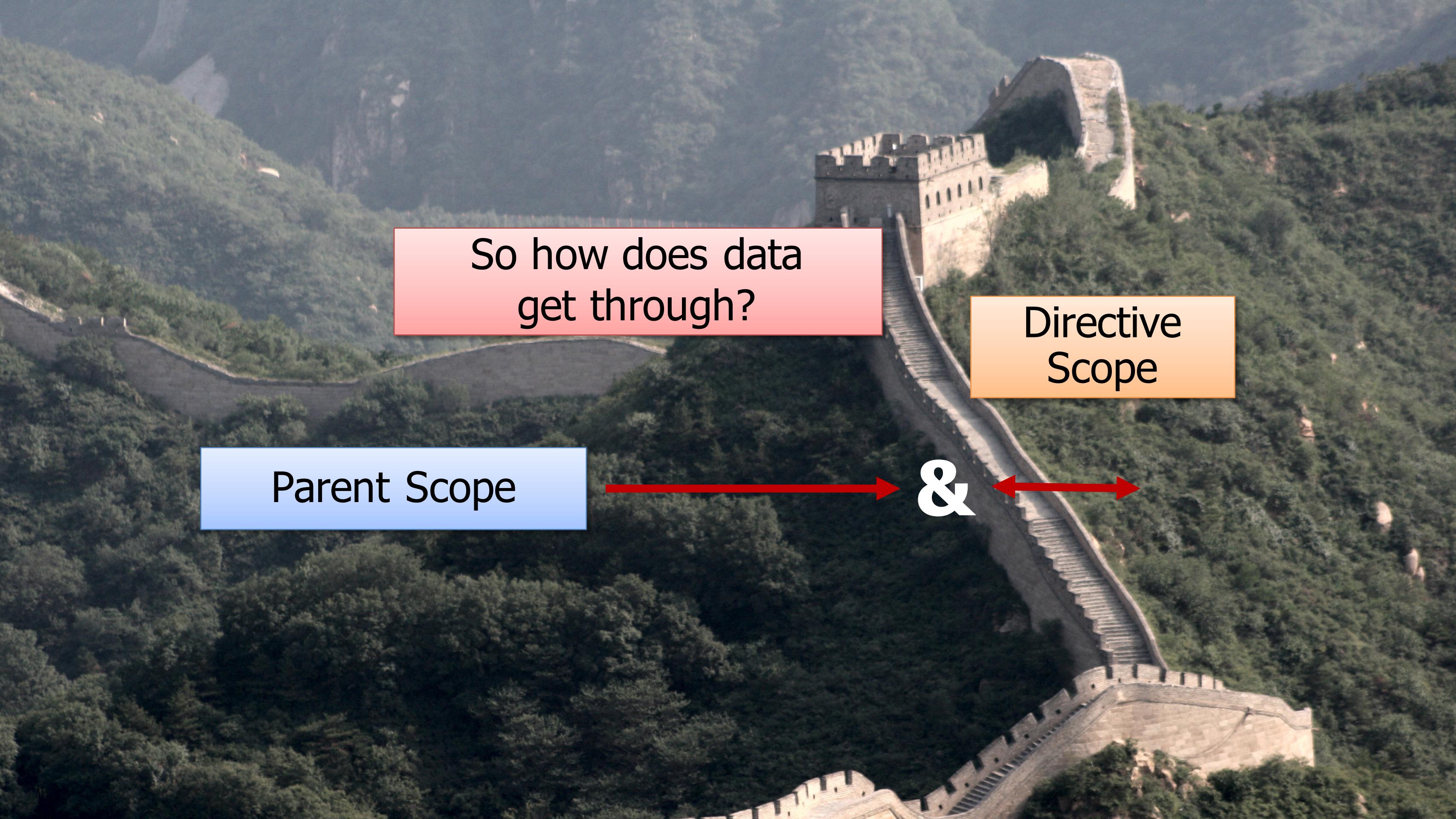
```
<my-directive customer="person" />
```

```
scope: { customer: '=' }
```

Directive

Two-way binding created

& Local Scope Property



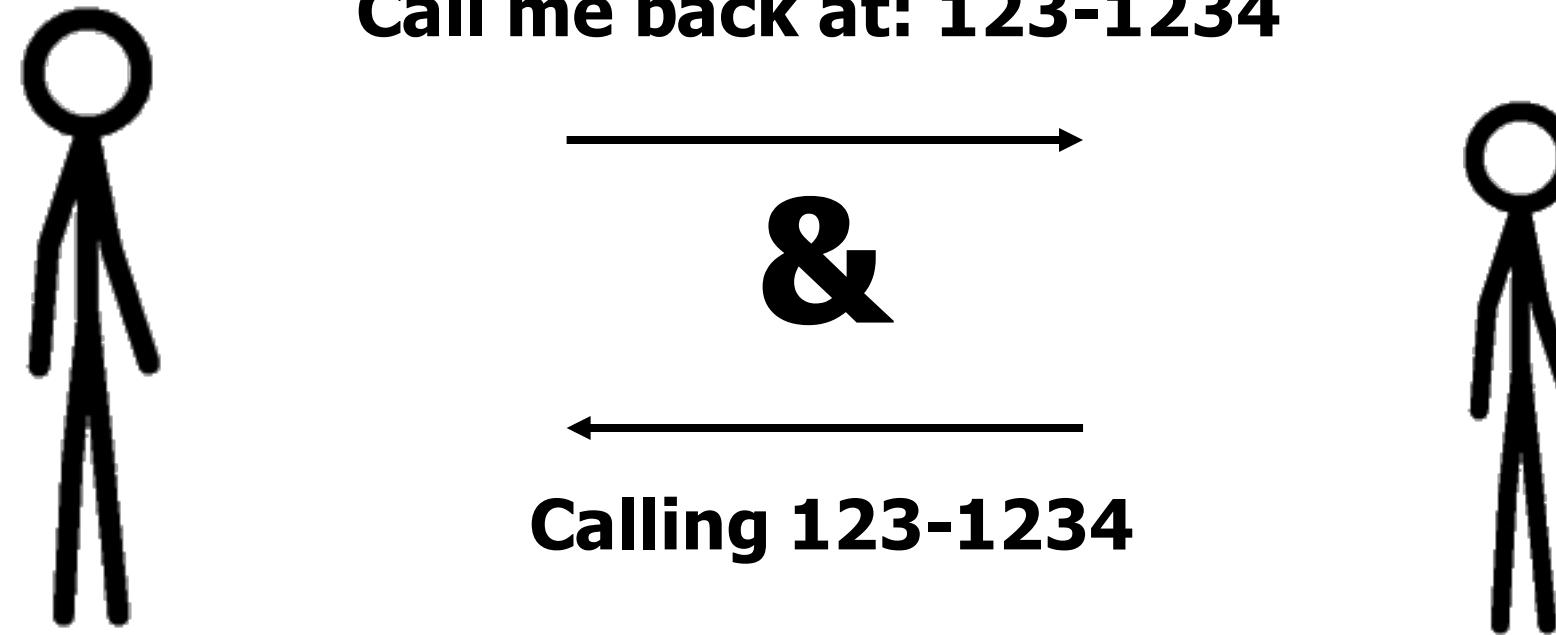
So how does data
get through?

Directive
Scope

Parent Scope



& Local Scope Property



Function Callback

& local scope property

```
$scope.click=function() {};
```

```
<my-directive action="click()" />
```

```
scope: { action: '&' }
```

Directive

Can invoke external function

The link() Function

Directive Categories

DOM-Driven Directives

All about DOM
Manipulation

Behavior-Driven Directives

All about handling DOM
Events

Data-Driven Directives

All about data, using
other directives and a
controller

The Link Function

```
angular.module('moduleName')
  .directive('myDirective', function () {
    return {
      restrict: 'EA',
      scope: {},
      template: '<div>{{ myVal }}</div>',
      controller: controller,
      link: function (scope, element, attrs) { }
    }
});
```

Link Function Parameters

```
(function () {
  var linkDemo = function () {
    return {
      link: function (scope, element, attrs) { }
    }
  };
});
```

The diagram illustrates the parameters of the link function with three callout boxes:

- An arrow points from the `scope` parameter to a blue box containing the text "Isolate or Shared Scope".
- An arrow points from the `element` parameter to a blue box containing the text "Attributes on element".
- An arrow points from the `attrs` parameter to a blue box containing the text "Element directive is on".

```
angular.module('moduleName')
  .directive('linkDemo', linkDemo);
}());
```

DOM Manipulation with jqLite

jqLite is a tiny, API-compatible subset of jQuery that allows Angular to manipulate the DOM.

<https://docs.angularjs.org/api/ng/function/angular.element>

Key jqLite Functions

angular.element()

find()

addClass()/css()

append()/remove()

attr()

html()/text()

on()/off()

ready()

Summary

Summary

@ Bind a local scope property to the value of a DOM attribute. The result is always a string.

```
scope: { name: '@' } → <my-directive name="{{name}}" />
```

= Creates a bi-directional binding between a local scope property and the parent scope property.

```
scope: { name: '=' } → <my-directive name="name" />
```

& Execute an expression/function in the context of the parent scope.

```
scope: { click: '&' } → <my-directive click="click()" />
```