AngularJS
What's the Big Deal?

Jim Duffy
TakeNote Technologies

Who Am I?

- Jim Duffy jduffy@takenote.com
- CEO/Founder TakeNote Technologies www.takenote.com
- Blog: www.geekswithblogs.net/takenote/
- Twitter: @jmduffy
- Microsoft Regional Director www.msrd.io
- 11 time Microsoft Most Valuable Professional (MVP)
- .NET, ASP.NET, HTML5, AngularJS & SQL Server Instructor, Mentor, Developer, and Consultant
- Experienced conference presenter
- Contributor to CODE Magazine

TakeNote TECHNOLOGIES

The Plan For This Session

- Why Use AngularJS
- AngularJS Overview
- Definitions
- Code Demos

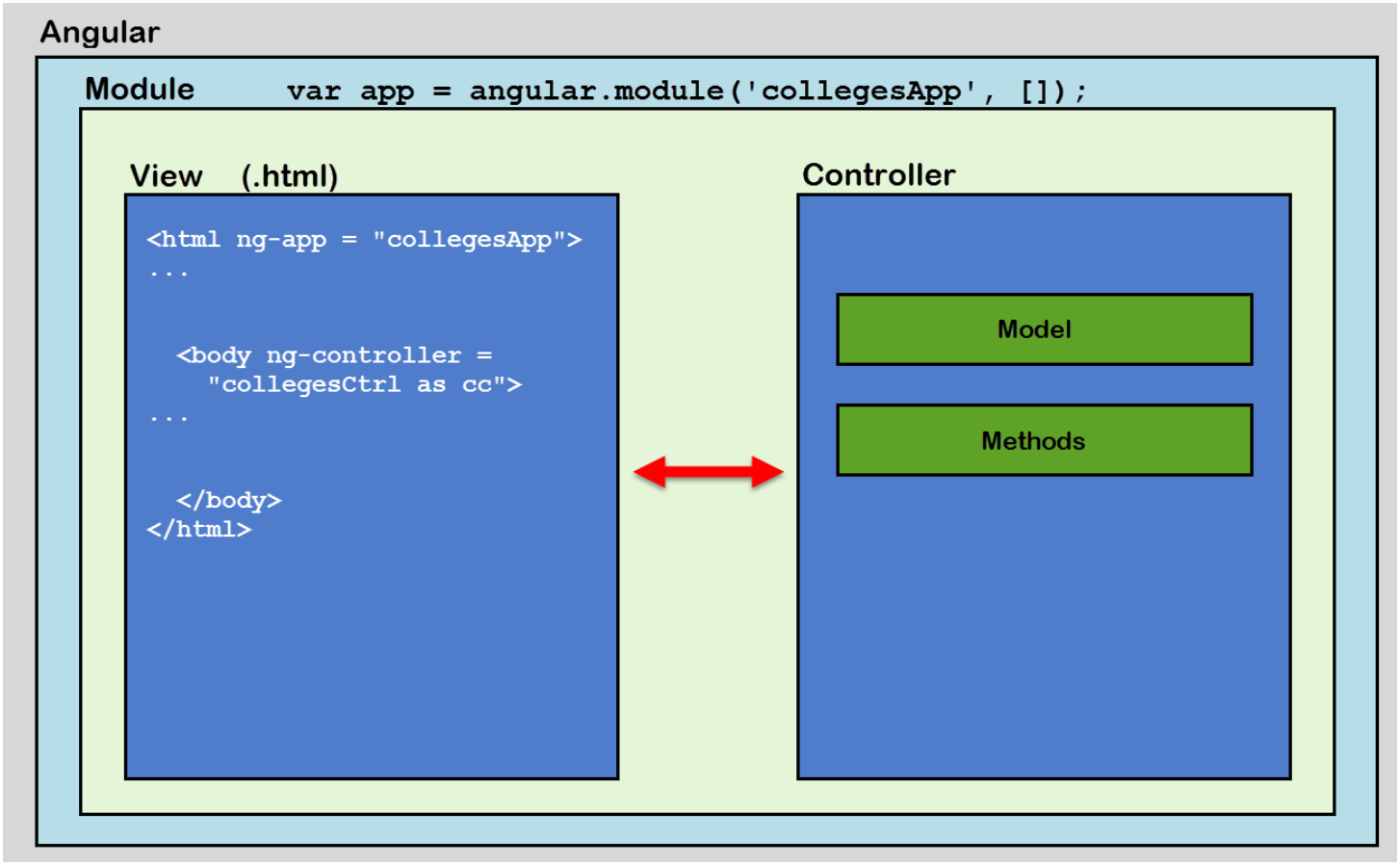
TakeNote TECHNOLOGIES

Why Use AngularJS?

- It's modular
- It provides powerful two-way data binding
- It uses expressive HTML via directives
- It is designed to be testable
- Built in route navigation

TakeNote TECHNOLOGIES

AngularJS At 30,000 Feet



TakeNote TECHNOLOGIES

Directives & Data Binding

- `ng-app`: defines the module
- One-way data binding
  - `ng-model`
- Two-way data binding
  - `{{ }}` syntax

TakeNote TECHNOLOGIES

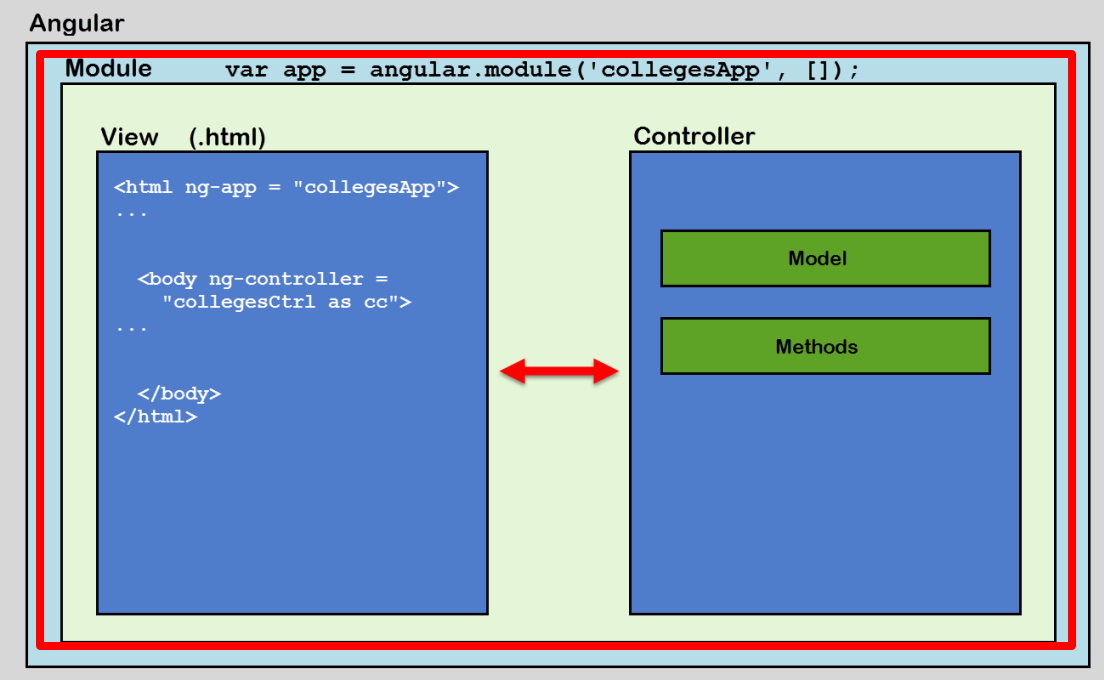
Directives & Data Binding

DEMO 1

TakeNote TECHNOLOGIES

Modules

- Modules define and contain the application



Creating A Module In app.js

- Using the `angular.module` function
- Use immediately invoked function expression (iffie)

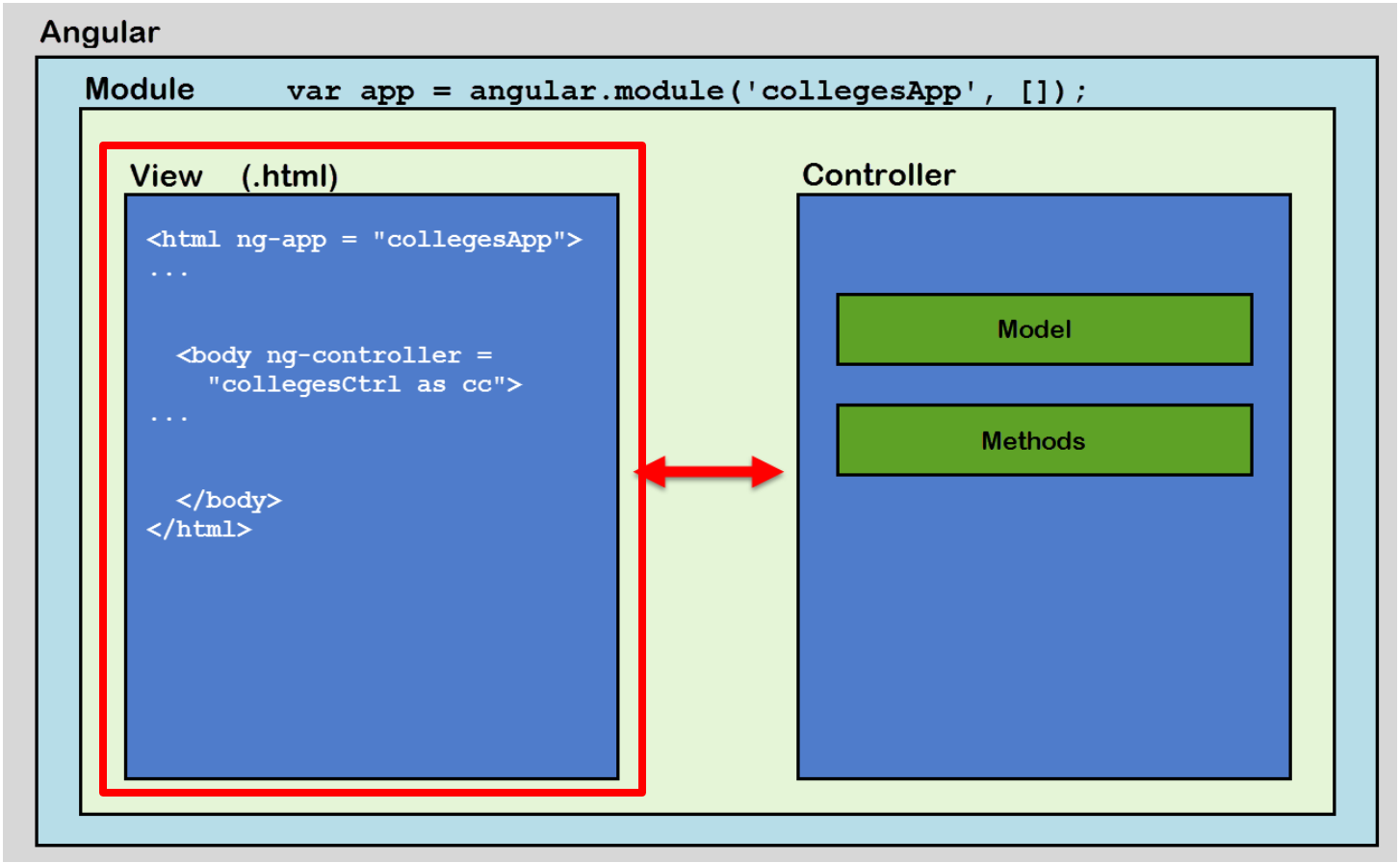
```
// app.js
(function () {
    "use strict";
    var app = angular.module("collegesApp", []);
}());
```

Modules

DEMO 2

TakeNote TECHNOLOGIES

Views



TakeNote TECHNOLOGIES

Applying Module To A View

- Using the ng-app directive

```
<html ng-app="collegesApp">
<head>
…
  <script src="scripts/angular.js"></script>
  <script src="app/app.js"></script>
</head>
```

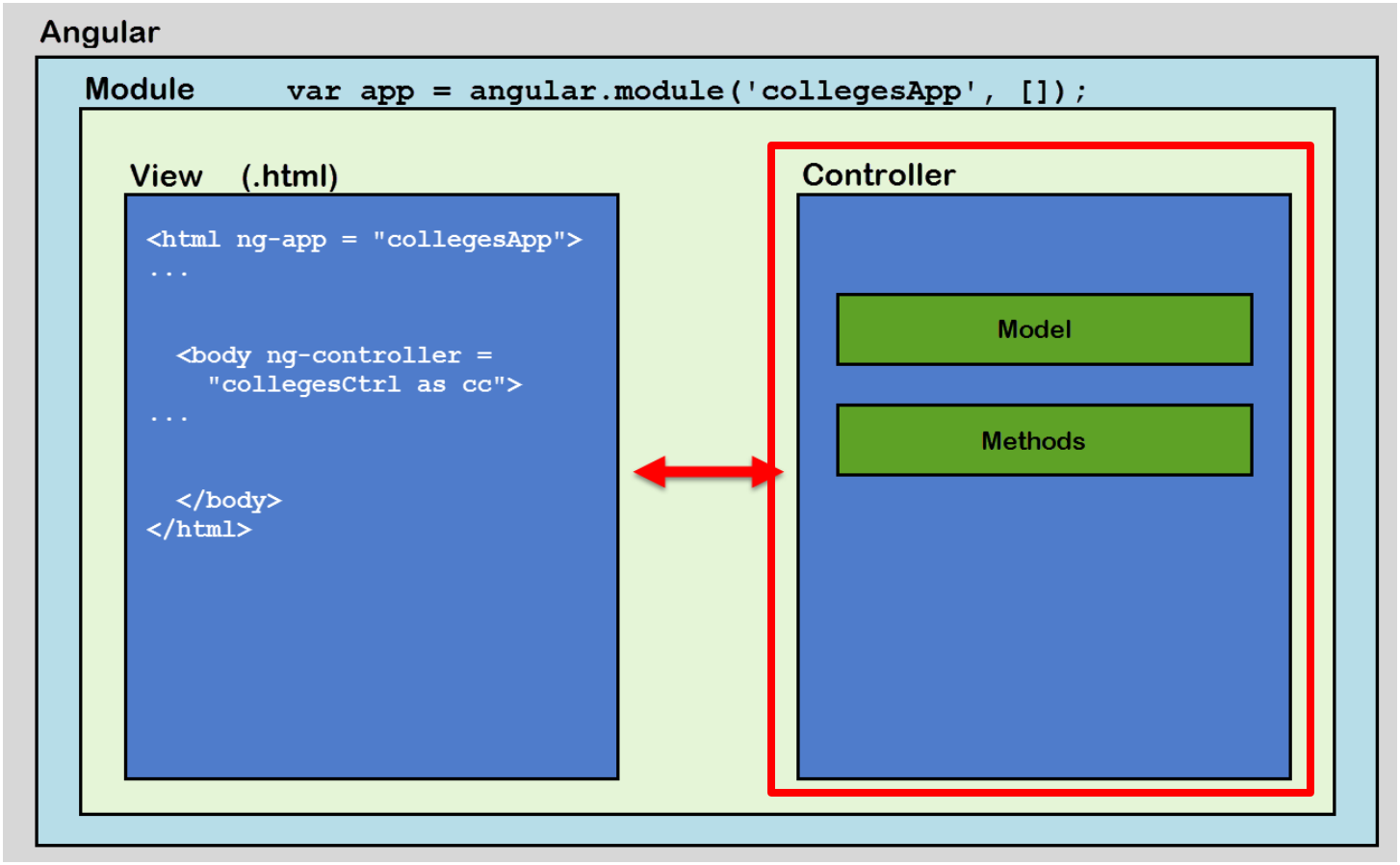

TakeNote TECHNOLOGIES

Apply Module To A View

DEMO 3

TakeNote TECHNOLOGIES

Controllers



TakeNote TECHNOLOGIES

Controller Code

```
// collegesCtrl.js
(function () {
    "use strict";
    angular.module("collegesApp")
        .controller("CollegesCtrl", CollegesCtrl);

    function CollegesCtrl ()
    {
      var vm = this;
      var propertyOne = "One";
      var propertyTwo = 2;
    }
}());
```


TakeNote TECHNOLOGIES

Wire Up A Controller In A View

- Using the ng-controller directive

```
<head>
      <title>Colleges</title>
…
      <script src="app/collegesCtrl.js"></script>
</head>
<body ng-controller="CollegesCtrl as cc">
    <div class="page-header">
        <h1>{{ cc.propertyOne }}'s Colleges</h1>
    </div>
…
```


TakeNote TECHNOLOGIES

Controller Best Practices

- Keep life simple and place each controller in it's own .js file
- Wrap controller in an immediately invoked function expression(iffe)
- Use either Controller or Ctrl as the name suffix (ex: ProductsCtrl or ProductsController)
- Use Pascal case when naming controller (ex: MyCoolController)

TakeNote TECHNOLOGIES

Controllers

DEMO 4

TakeNote TECHNOLOGIES

Defining A Data Model

```
function CollegesCtrl ()
    {
…
     vm.colleges = {
       conference: "ACC",
       items: [{ university: "University of Miami",
                   city: "Coral Gables",
                   founder: false },
                 { university: "North Carolina",
                   city: "Chapel Hill",
                   founder: true }]
     };
…
```


TakeNote TECHNOLOGIES

Iterating Through Data

- Using the ng-repeat directive

```
<tbody>
      <tr ng-repeat="college in cc.colleges.items">
          <td>{{ college.name }}</td>
          <td>{{ college.city }}</td>
          <td>{{ college.founder }}</td>
      </tr>
</tbody>
```


TakeNote TECHNOLOGIES

Defining A Data Model

DEMO 5

TakeNote TECHNOLOGIES

Expressions

- JavaScript-like code snippets used in bindings such as {{ expression }}
- Examples
  - 1+2={{1+2}}
  - user.name

```
<span> 1+2={{ 1+2 }} </span>
<span class="label label-default">
      {{vm.colleges.items.length}}
</span>
```


TakeNote TECHNOLOGIES

Filters

- Filters are implemented with |

```
<tr ng-repeat="college in cc.colleges.items |
       orderBy: 'name' | filter: 'State' ">
    <td>{{ college.name | uppercase }}</td>
    <td>{{ college.city | lowercase }}</td>
```


TakeNote TECHNOLOGIES

Expressions & Filters

DEMO 6

TakeNote TECHNOLOGIES

Events

- Using the ng-click event

```
<div class="input-group">
   <input class="form-control"
      ng-model="vm.collegename" />

   <span class="input-group-btn">
      <button type="button"
            class="btn btn-primary"
            ng-click="vm.addCollege()">Add</button>
   </span>
</div>
```


TakeNote TECHNOLOGIES

Events

DEMO 7

TakeNote TECHNOLOGIES

Routes

- Routes are the key to single-page applications (SPA)
- Routes control which page will be displayed in the host page
- Used to bind the controller to the view
- Require additional angular-route.js file

```
//app.js
var app = angular.module('collegesApp', ['ngRoute']);
app.config(function ($routeProvider) {
   $routeProvider.when('/',
     {
         templateUrl: '/partials/colleges.html',
         controller: 'CollegesCtrl',
         controllerAs: 'vm'
     });
   $routeProvider.when('/colleges',
     {
         templateUrl: '/partials/colleges.html',
         controller: 'CollegesCtrl', controllerAs: 'vm'
     });
   $routeProvider.when('/dataentry',
     {
         templateUrl: '/partials/dataentry.html',
         controller: 'DataEntryCtrl', controllerAs: 'vm'
          });
   $routeProvider.otherwise({ redirectTo: '/' });
});
```


TakeNote TECHNOLOGIES

Routes

DEMO 8

TakeNote TECHNOLOGIES

Data Entry Forms

- Visibility attributes
  - `ng-show`: Displays an element
  - `ng-hide`: Hides an element
- Data validation attributes
  - `ng-minlength`: min string length
  - `ng-maxlength`: max string length
  - `ng-pattern`: regex comparison

TakeNote TECHNOLOGIES

Data Entry Forms

- Properties
  - `$pristine`
  - `$dirty`
  - `$valid`
  - `$invalid`
  - `$error`

TakeNote TECHNOLOGIES

```
<div class="form-group">
   <label for="firstname">First Name:</label>
   <input type="text" name="firstname" autofocus required
      placeholder="Enter first name"
      ng-model="firstname" ng-minlength="2"
      ng-maxlength="20" ng-pattern="/^[a-z]+$/"
      class="form-control" />
   <div class="error-container"
       ng-show="myForm.firstname.$dirty &&
                myForm.firstname.$invalid">
       <small class="error"
          ng-show="myForm.firstname.$error.required">
             First name is required.</small>
       <small class="error"
          ng-show="myForm.firstname.$error.minlength">
             First name requires at least 2 char.</small>
       <small class="error "
          ng-show="myForm.firstname.$error.maxlength">
             First Name cannot exceed 20 chars.</small>
   </div>
</div>
```


TakeNote TECHNOLOGIES

Data Entry Forms

DEMO 9

TakeNote TECHNOLOGIES

Data Entry Forms - CSS

- `ng-pristine`: Elements the user has not interacted are added to this class.
- `ng-dirty`: Elements the user has interacted with are added to this class.
- `ng-valid`: Elements that are valid are in this class.
- `ng-invalid`: Elements that are not valid are in this class.

TakeNote TECHNOLOGIES

Data Entry Forms – CSS

```
form .ng-pristine {
      background-color: #fffeb9;
}
form .ng-valid.ng-dirty {
      background-color: lightgreen;
}
form .ng-invalid.ng-dirty {
      background-color: #ff0000;
}
```


TakeNote TECHNOLOGIES

Data Validation

DEMO 10

TakeNote TECHNOLOGIES

Resources

- https://www.angularjs.org/
- https://github.com/angular/angular.js
- https://docs.angularjs.org/guide
- https://blog.angularjs.org/
- www.takenote.com

Hands On Training Classes from TakeNote Technologies

AngularJS Fundamentals (next class 12/15 & 12/16)

Developing Line of Business Applications With AngularJS (next class 12/17 & 12/18)

$99 Discount code: TRINUGSS

TakeNote TECHNOLOGIES

Thank You for Attending!

- My contact info:

Jim Duffy
jduffy@takenote.com
CEO/Founder
TakeNote Technologies
www.takenote.com
Twitter: @jmduffy