AngularJS

UNIVERSITY OF
TORONTO

What is AngularJS

→ Javascript Framework
→ MVC
→ for Rich Web Application Development
→ by Google

UNIVERSITY OF TORONTO

Why AngularJS

"Other frameworks deal with HTML's shortcomings by either abstracting away HTML, CSS, and/or JavaScript or by providing an imperative way for manipulating the DOM. Neither of these address the root problem that HTML was not designed for dynamic views".

- Lightweight (< 36KB compressed and minified)
- Free
- Separation of concern
- Modularity
- Extensibility & Maintainability
- Reusable Components

UNIVERSITY OF TORONTO

What we used previously.

- Allows for DOM Manipulation
- Does not provide structure to your code
- Does not allow for two way binding

UNIVERSITY OF TORONTO

What we used previously.

UNIVERSITY OF TORONTO

```
<p>Select value: {{ theValue }}</p>
<select ng-model="theValue">
  <option value="1">1</option>
  <option value="2">2</option>
</select>
```

```
<p>Select value: <span
id="theValue"></span></p>
<select id="theSelect">
  <option value="1">1</option>
  <option value="2">2</option>
</select>

$(function() {
  $('#theSelect').on('change',
function() {
    var value = $(this).val();
    $('#theValue').text(value);
  }
});
```

UNIVERSITY OF TORONTO

Some Features

→ Two-way Data Binding – Model as single source of truth
→ Directives – Extend HTML
→ MVC
→ Dependency Injection
→ Testing
→ Deep Linking (Map URL to route Definition)
→ Server-Side Communication

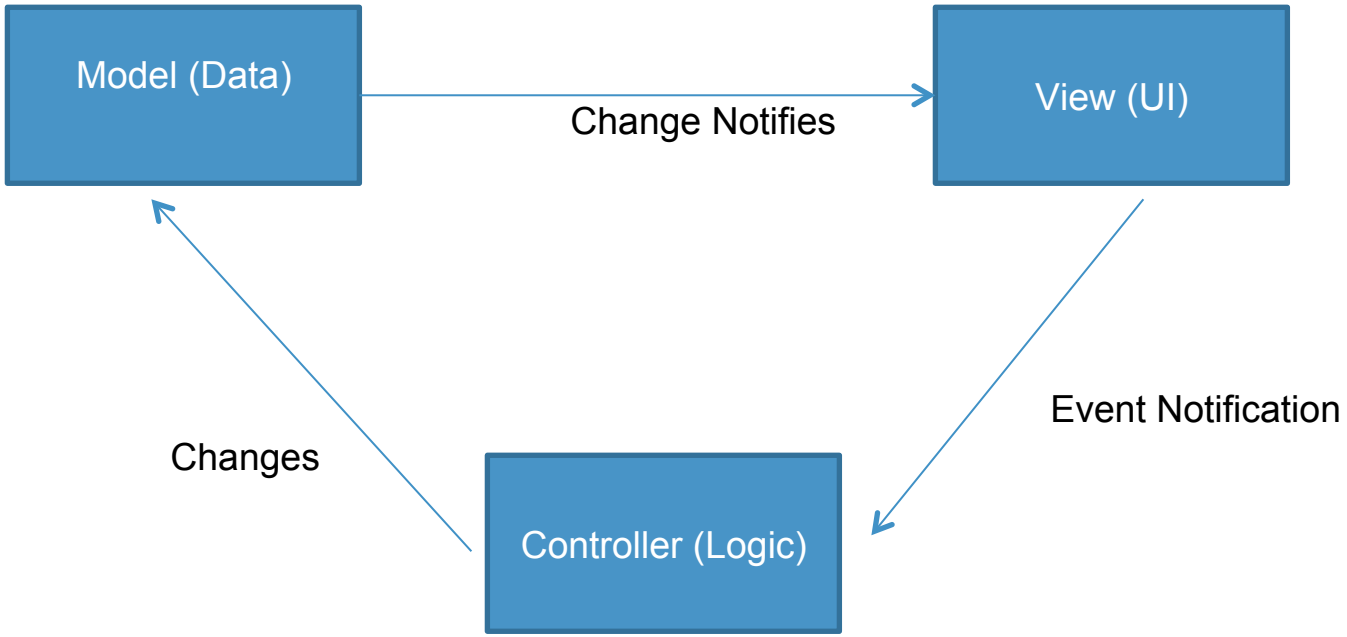
UNIVERSITY OF TORONTO

Simple Example

```
<html ng-app>
<head>
  <script src='angular.min.js'></script>
</head>
<body>
  <input ng-model='user.name'>
  <div ng-show='user.name'>Hi
{{user.name}}</div>
</body>
</html>
```

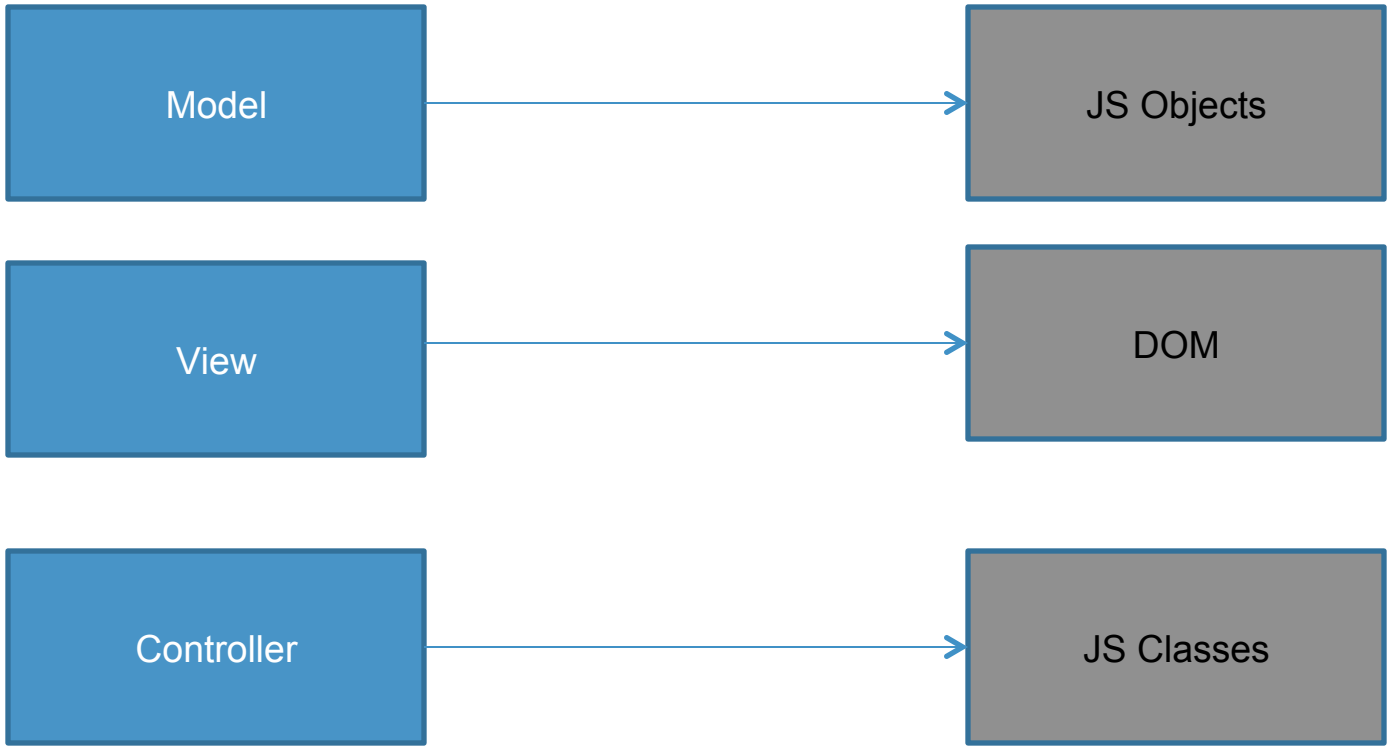
UNIVERSITY OF TORONTO

MVC



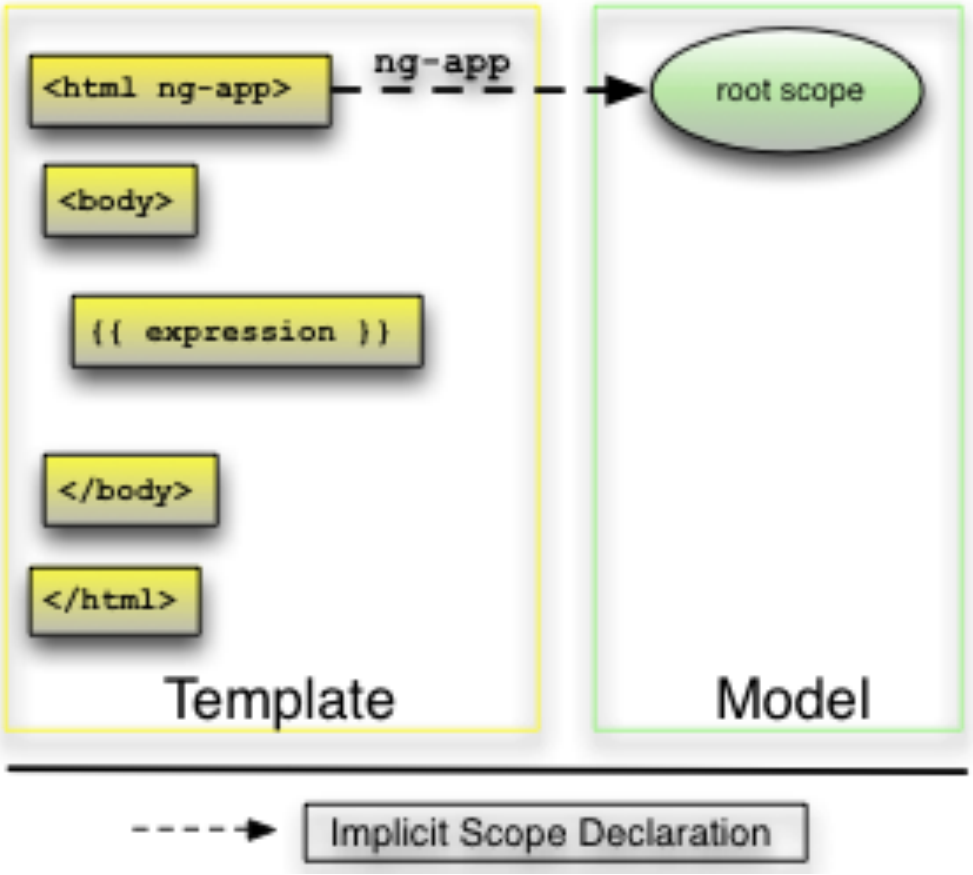
UNIVERSITY OF TORONTO

MVC



UNIVERSITY OF TORONTO

Bootstrapping



UNIVERSITY OF TORONTO

Expressions

AngularJS binds data to HTML using Expressions.

```
<div>
        <p>My first expression: {{ 5 + 5 }}</p>
</div>

<div ng-app="" ng-init="points=[1,15,19,2,40]">
<p>The third item is <span ng-bind="points[2]"></span></p>
</div>
```

UNIVERSITY OF TORONTO

Directives

AngularJS lets you extend HTML with new attributes called Directives.

```
<div ng-app="" ng-init="firstName='John'">

<p>Name: <input type="text" ng-model="firstName"></p>
<p ng-bind="firstName"></p>
</div>
```

https://docs.angularjs.org/api/ng/directive

UNIVERSITY OF TORONTO

Directives

```
<div ng-app="" ng-init="names=[
{name:'Jani',country:'Norway'},
{name:'Hege',country:'Sweden'},
{name:'Kai',country:'Denmark'}]">

<ul>
  <li ng-repeat=”x in names">
    {{ x.name + ', ' + x.country }}
  </li>
</ul>

</div>
```

UNIVERSITY OF TORONTO

Controllers

AngularJS controllers *control the data* of AngularJS applications.

```
<div ng-app="myApp" ng-controller="personCtrl">

First Name: <input type="text" ng-
model="firstName"><br>
Last Name: <input type="text" ng-
model="lastName"><br>
<br>
Full Name: {{fullName()}}

</div>

<script>
var app = angular.module('myApp', []);
app.controller('personCtrl', function($scope) {
    $scope.firstName = "John";
    $scope.lastName = "Doe";
    $scope.fullName = function() {
        return $scope.firstName + " " + $scope.lastName;
    }
});
</script>
```

UNIVERSITY OF TORONTO

Filters

Filter is used to do some transformation of the scoped Data. Filter is applied using the symbol | (pipe).

Filter	Description
currency	Format a number to a currency format.
filter	Select a subset of items from an array.
lowercase	Format a string to lower case.
orderBy	Orders an array by an expression.
uppercase	Format a string to upper case.

```
<div ng-app="myApp" ng-controller="costCtrl">

<input type="number" ng-model="quantity">
<input type="number" ng-model="price">

<p>Total = {{ (quantity * price) | currency }}</p>

</div>
```

UNIVERSITY OF TORONTO

Server Interaction

$http is an AngularJS service for reading data from remote servers.

```
{
"records": [
  {
    "Name" : "Alfreds Futterkiste",
    "City" : "Berlin",
    "Country" : "Germany"
  },
  {
    "Name" : "Berglunds snabbköp",
    "City" : "Luleå",
    "Country" : "Sweden"
  },
  {
    "Name" : "Centro comercial Moctezuma",
    "City" : "México D.F.",
    "Country" : "Mexico"
  }
]}
```

UNIVERSITY OF TORONTO

Server Interaction

$http is an AngularJS service for reading data from remote servers.

```
<div ng-app="myApp" ng-controller="customersCtrl">
<ul>
  <li ng-repeat="x in names">
    {{ x.Name + ', ' + x.Country }}
  </li>
</ul>
</div>

<script>
var app = angular.module('myApp', []);
app.controller('customersCtrl', function($scope, $http) {
    $http.get("http://www.w3schools.com/angular/customers.php")
    .success(function(response) {$scope.names = response.records;});
});
</script>
```

UNIVERSITY OF TORONTO

Modules

A module is a container for the different parts of an application. All application controllers should belong to a module.

```
<body>
<div ng-app="myApp" ng-controller="myCtrl">
{{ firstName + " " + lastName }}
</div>

<script>
var app = angular.module("myApp", []);
app.controller("myCtrl", function($scope) {
    $scope.firstName = "John";
    $scope.lastName = "Doe";
});
</script>
</body>
```

UNIVERSITY OF TORONTO

$Scope

"Scope is an object that refers to the application model. It is an execution context for expressions. Scopes are arranged in hierarchical structure which mimic the DOM structure of the application. Scopes can watch expressions and propagate events." - From Angular website

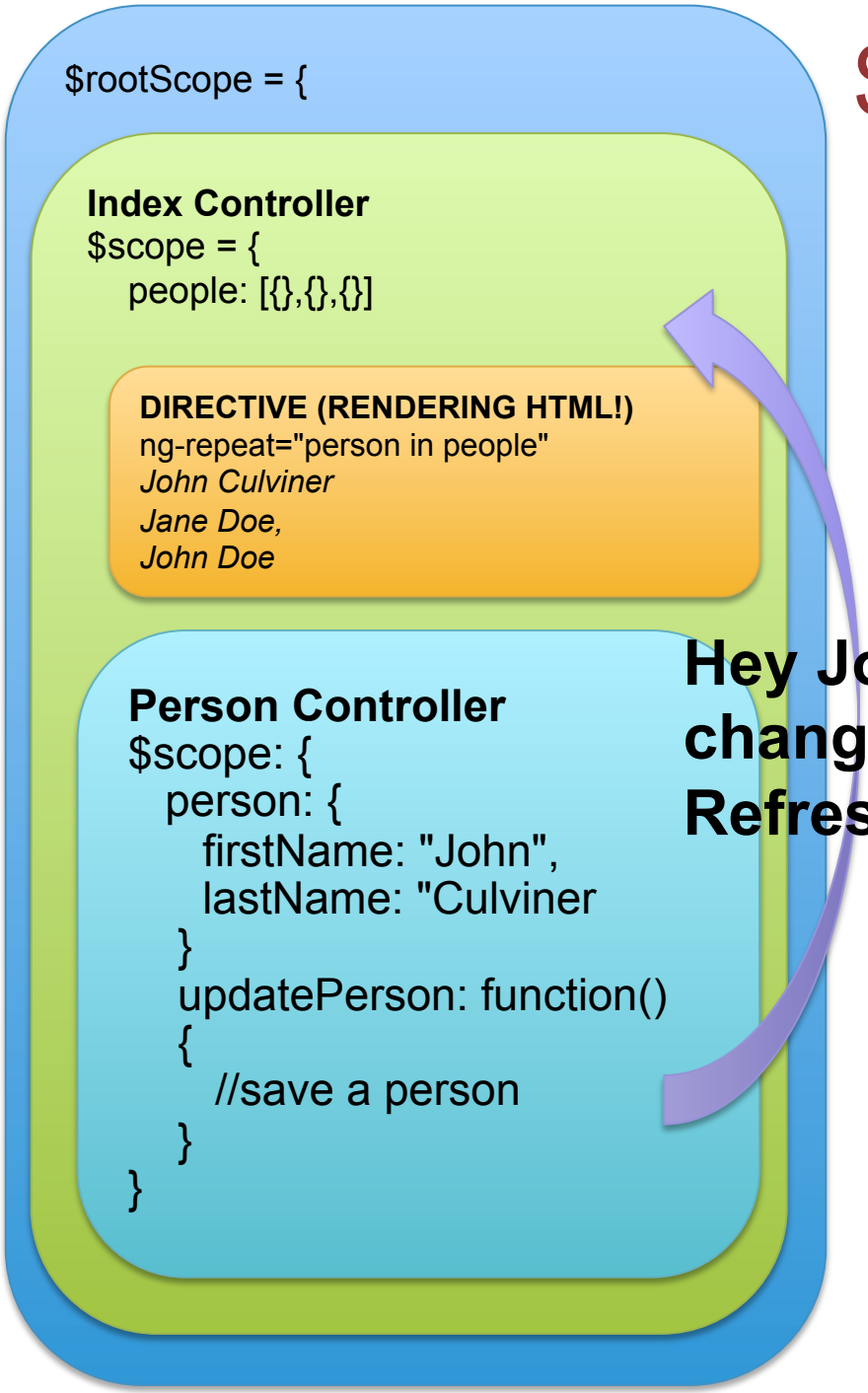
→ {{firstName + " " + lastName}} is an expression executed within scope

→ Scope can be hierarchal with DOM nesting of directives

→ Watches can be used to watch for changes to scope ex:

```
$scope.$watch("firstName", function(value) {
        //update the DOM with the new value
});
```

UNIVERSITY OF TORONTO



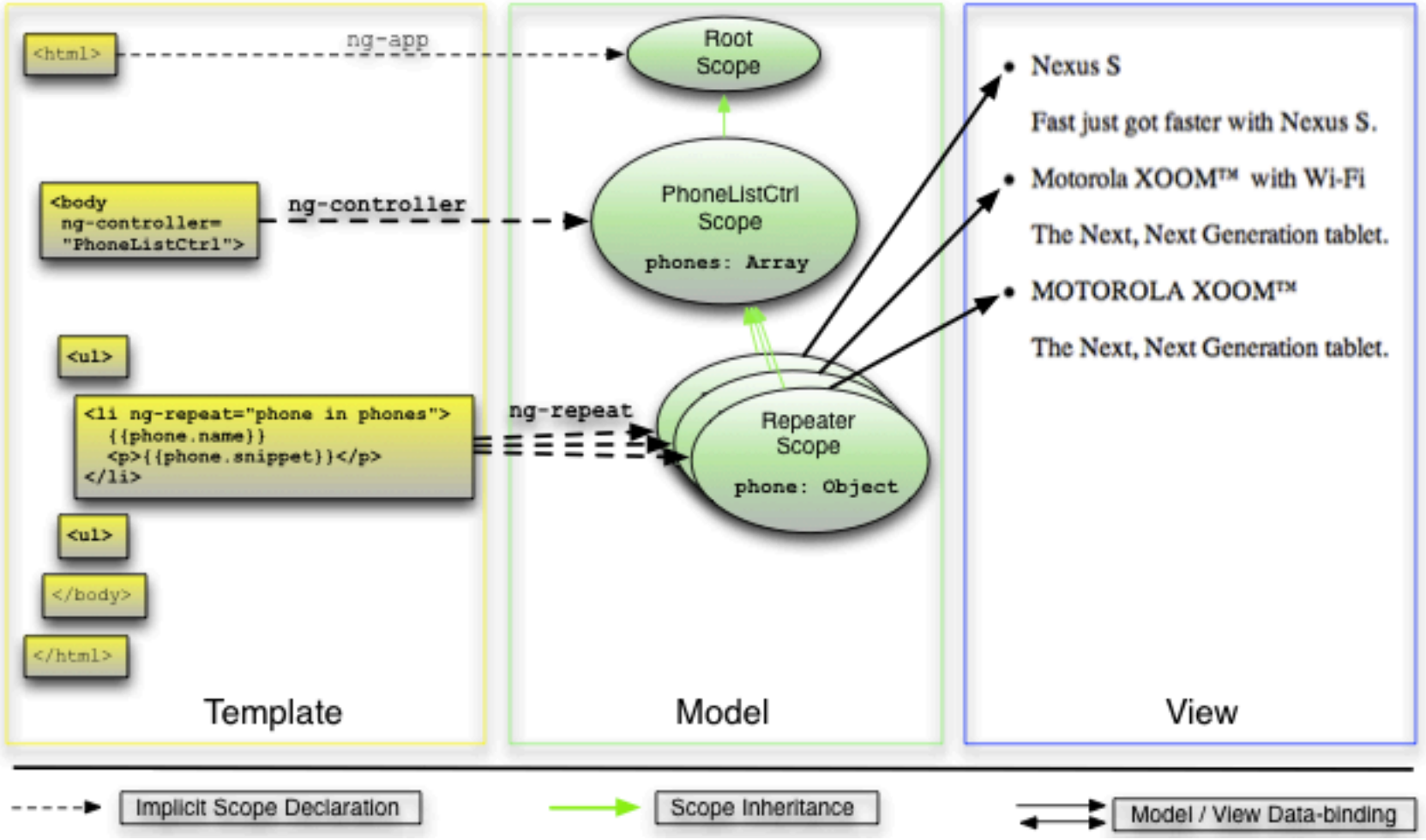
$scope .emit/.on

- Scopes can "message" parent/child scopes
  - $scope.$emit(…)

    Message upward
  - $scope.$broadcast(…)

    Message downward
- Here:
  - When a person changes
  - Notify the "Index" controller to refresh it's list (which has now changed)

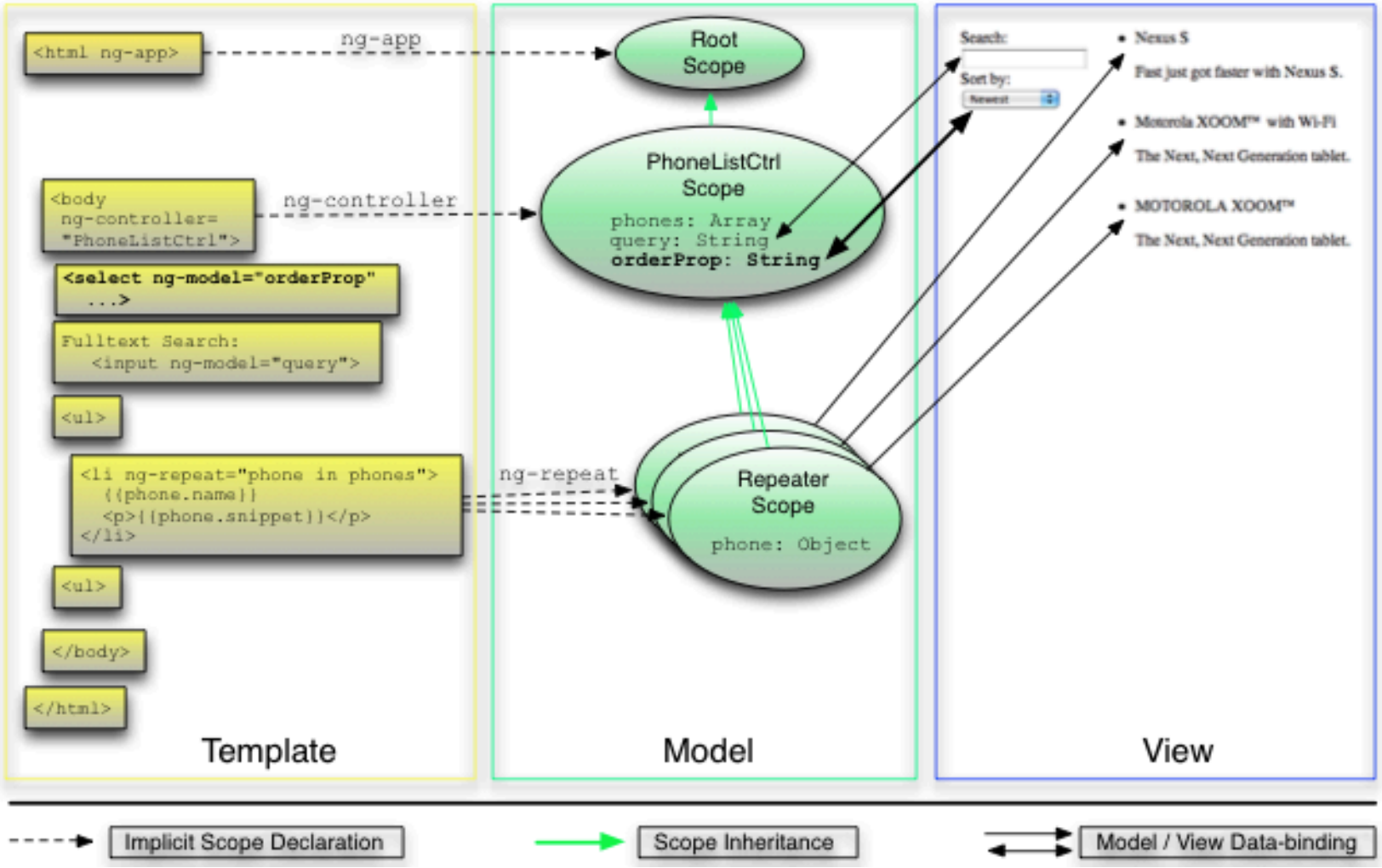
Putting it on Together



Contents taken from https://docs.angularjs.org/tutorial/step_04

UNIVERSITY OF TORONTO

Two way data Binding



UNIVERSITY OF TORONTO

$ngRoutes

“Application routes in Angular are declared via the $routeProvider, which is the provider of the $route service. This service makes it easy to wire together controllers, view templates, and the current URL location in the browser. Using this feature, we can implement deep linking, which lets us utilize the browser's history (back and forward navigation) and bookmarks.” - Angularjs.org

```
var phonecatApp = angular.module(’phonecatApp', [
  'ngRoute',
  ’OtherDepedencies'
]);
```

UNIVERSITY OF TORONTO

$ngRoutes

```
phonecatApp.config(['$routeProvider',
  function($routeProvider) {
    $routeProvider.
      when('/phones', {
        templateUrl: 'partials/phone-list.html',
        controller: 'PhoneListCtrl'
      }).
      when('/phones/:phoneId', {
        templateUrl: 'partials/phone-detail.html',
        controller: 'PhoneDetailCtrl'
      }).
      otherwise({
        redirectTo: '/phones'
      });
  }]);
```

UNIVERSITY OF TORONTO

$ngRoutes

….

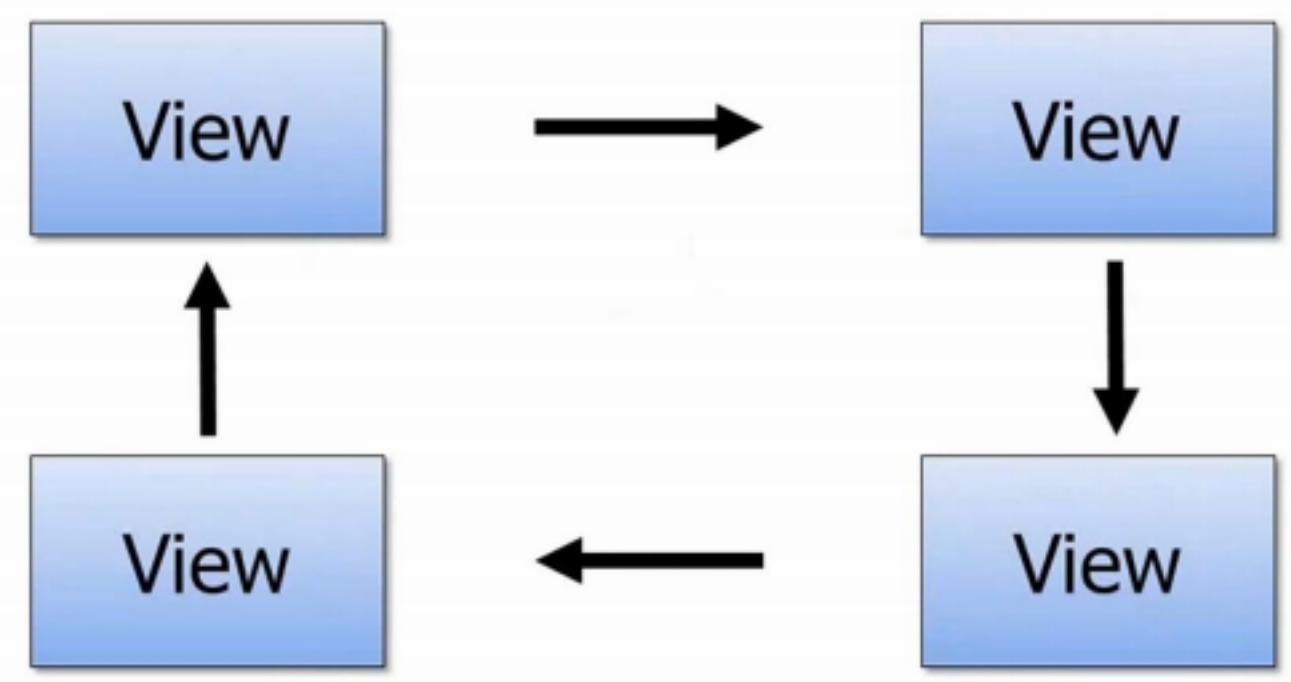
```
<ul class="phones">
        <li ng-repeat="phone in phones | filter:query | orderBy:orderProp"
class="thumbnail">
          <a href="#/phones/{{phone.id}}" class="thumb"><img ng-
src="{{phone.imageUrl}}"></a>
          <a href="#/phones/{{phone.id}}">{{phone.name}}</a>
          <p>{{phone.snippet}}</p>
        </li>
      </ul>
```

….

Full example can be found in https://docs.angularjs.org/tutorial/step_07

UNIVERSITY OF TORONTO

Single Page application



http://fastandfluid.com/publicdownloads/
AngularJSIn60MinutesIsh_DanWahlin_May2013.pdf

UNIVERSITY OF TORONTO

Single Page application

The Challenge with SPAs

DOM Manipulation

History

Module Loading

Routing

Caching

Object Modeling

Data Binding

Ajax/Promises

View Loading

UNIVERSITY OF TORONTO