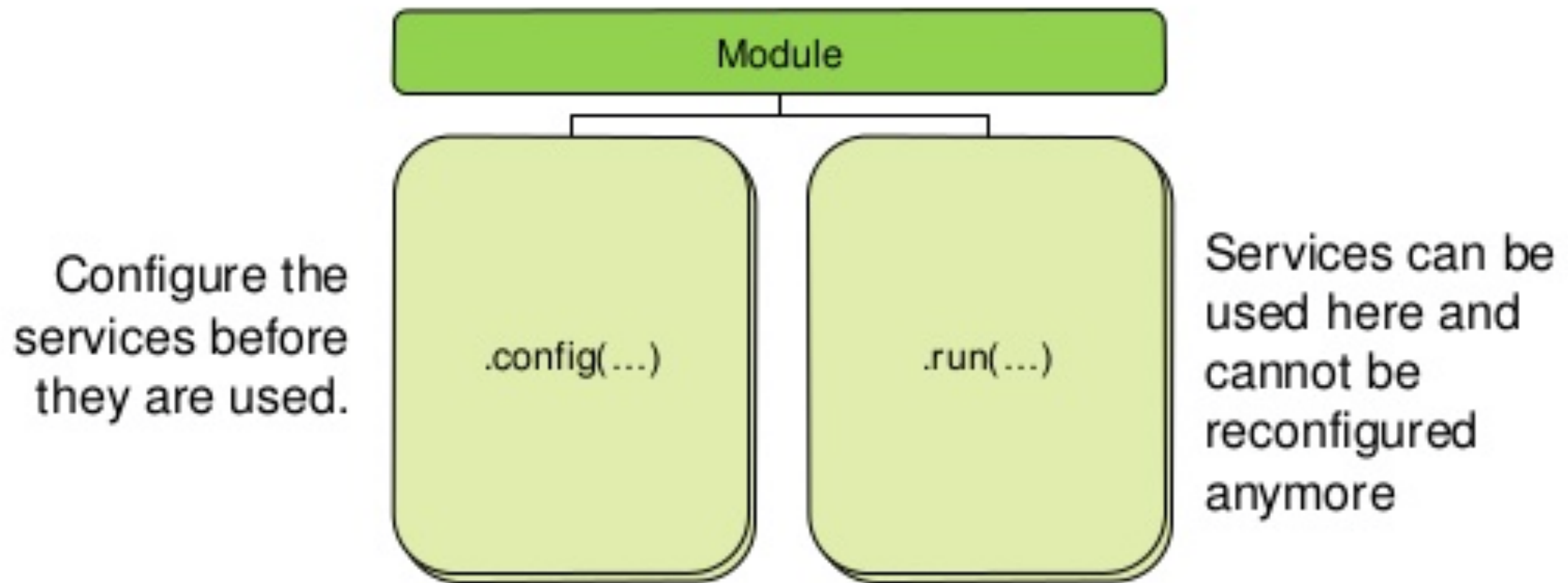


AngularJS Sección:

- ✦ Events: \$on, \$broadcast, \$rootScope
- ✦ Inputs: -ng-model -ng-check -ng-value
- ✦ Servicios: services, factory, value
- ✦ Directive
- ✦ ng-show , \$log, \$http, \$timeout
- ✦ Formas de crear controladores y servicios

✧ Fases de un **Modulo**

Two phases when initializing a module



Configuramos Urls,
variables, funciones,
etc

Utilizamos los
servicios ya
instanciados

AngularJS Directive

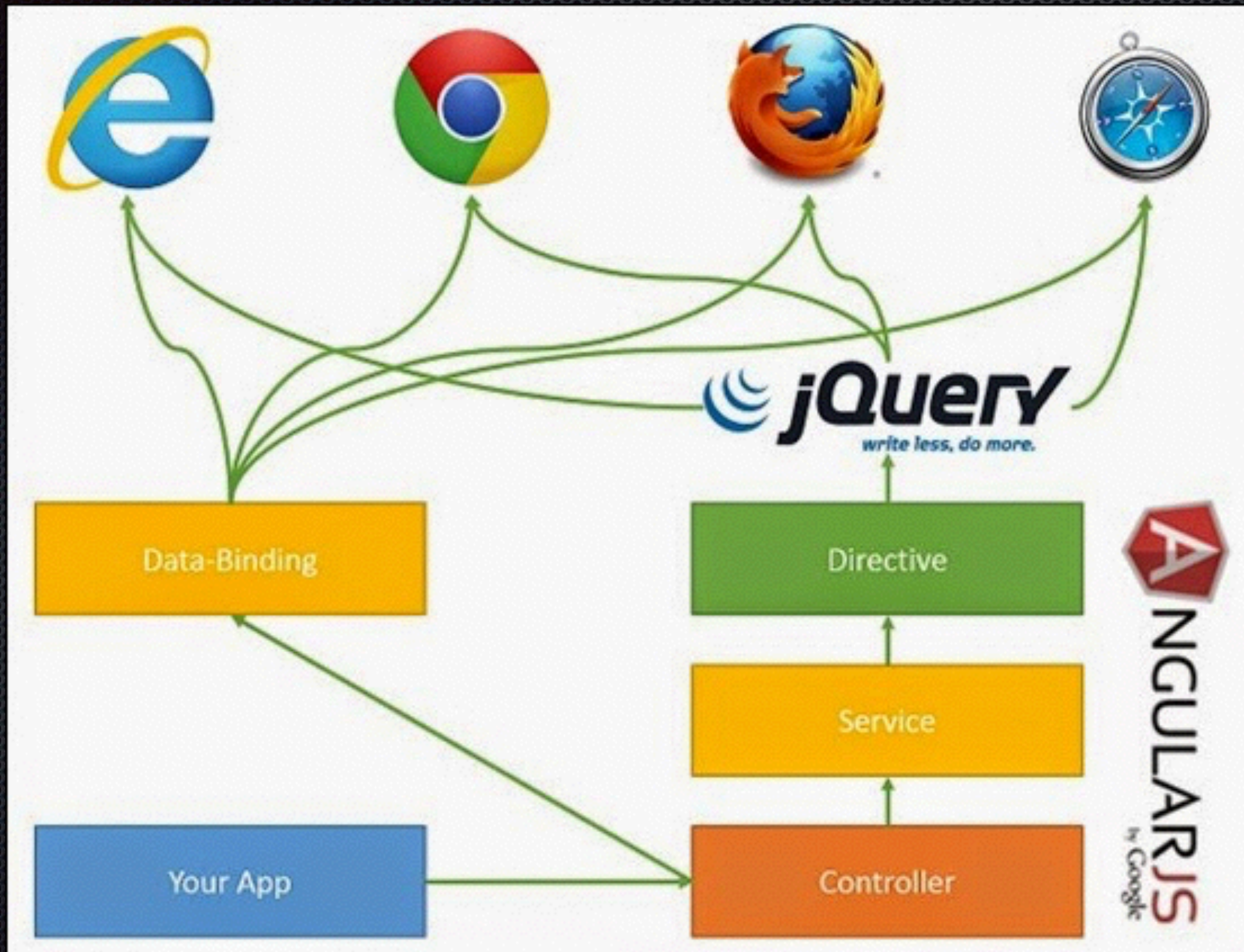
- ✦ Extensión del HTML, podemos crear propias y utilizar las que vienen en angular

```
<my-widget title="Another Widget">
  This is some content of my other widget ...
</my-widget>
```

- **transclude**
- **link**
- **template**
- **replace**
- **restrict**

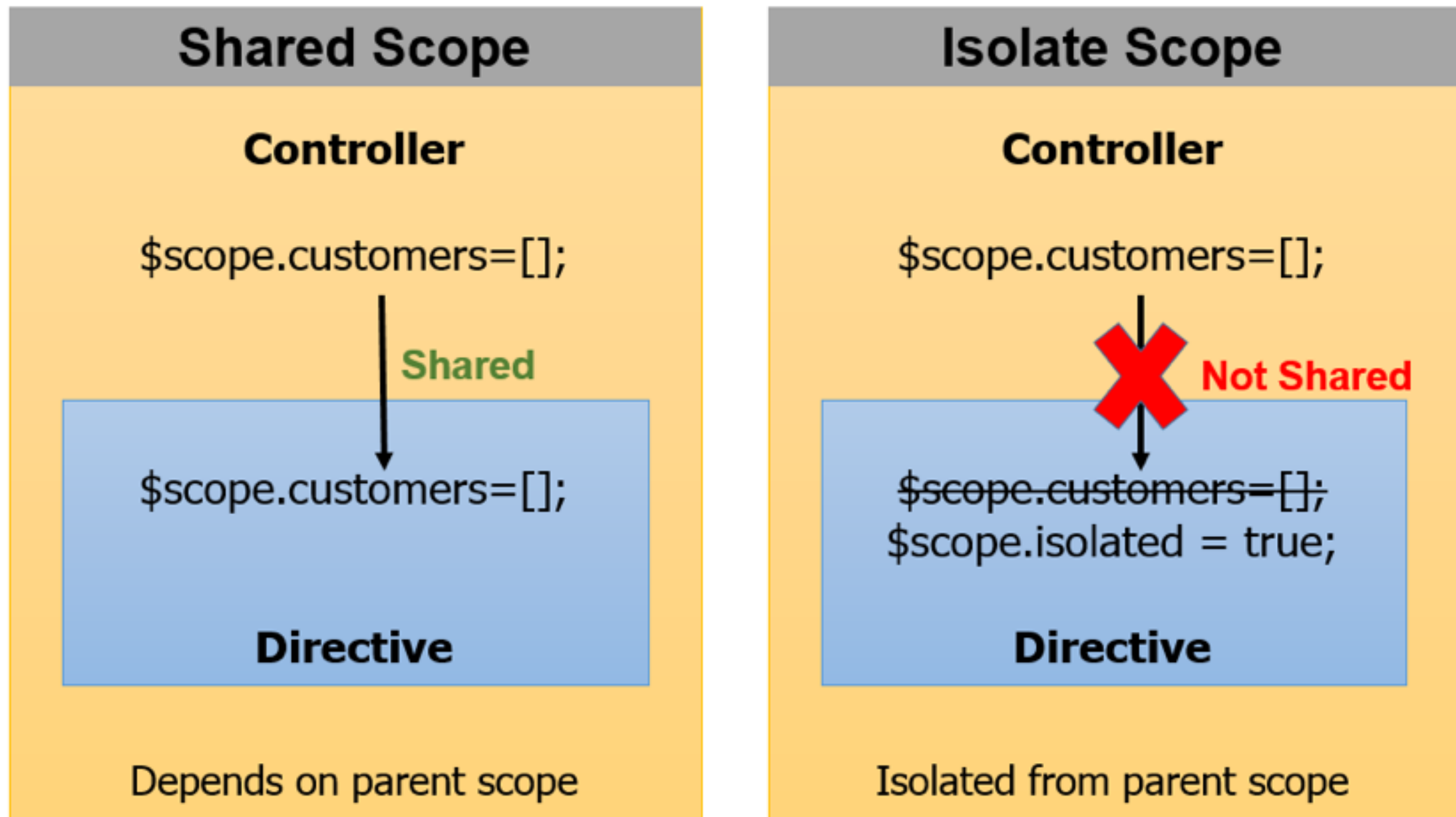
```
angular.module('myApp', []).
  directive('myWidget', function () {
    return {
      restrict: 'E',
      replace: true,
      transclude: true,
      template: '<div class="widget">' +
        '<div class="content" ng-transclude></div>' +
        '</div>',
      link: function (scope, element, attrs) {
        element.prepend("<div class='title'>" + attrs.title + "</div>");
      }
    };
  });
```


AngularJS Directive, DOM Manager



AngularJS Directive - Compartir y Aislar

Shared and Isolate Scope



AngularJS Directive - LocalScope

@ or @attr

```
$scope.name='James';
```

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

scope: { name: '@' }

Directive

Directive can access a string value

= or =attr

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

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

scope: { customer: '=' }

Directive

Two-way binding created

& or &attr

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

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

scope: { action: '&' }

Directive

Can invoke external function

Plantilla interna

Directiva **ngHtml5Dir** , scope impresión: *Scope controlador* <<*DirectivasCtrl*>>

Replace HTML Directiva

Directiva ngReplaceDir scope impresión: *Scope controlador* <<*DirectivasCtrl*>>

Directive restricción E

elementDir

Directive restricción A

attributeDir: valor por parámetro

Directive con controlador - aislada

Nombre de ítem

add

0. Ítem: **uno**

1. Ítem: **dos**

2. Ítem: **tres**

Directive con atributos: @ & =

Plantilla interna

Directiva `ngHTemplateDir` , scope impesión: *Scope controlador* `<<DirectivasCtrl>>`

```
<div ngh-template-dir>
  <strong>Todo esto desaparece</strong>
  <div>
    No existe
  </div>
</div>
```

```
.directive('ngHTemplateDir', function () {
  return {
    template: 'Directiva <strong>ngHTemplateDir</strong> , scope impesión: <em>{{myScopeVar}}</em>'
  };
})
```


Replace HTML Directiva

Directiva `ngReplaceDir` scope impresión: *Scope controlador <<DirectivasCtrl>>*

```
.directive('ngReplaceDir', function () {
  return {
    replace: true,
    template:
      '<div style="font-size: 18px;" >\n\n' +
      'Directiva <strong>ngReplaceDir</strong> scope impresión: <em>{{myScopeVar}}</em>\n\n' +
      '</div>',
    link: function (scope, element, attrs)
    {
      var mouseover = function() {
        element.css( "color", "red" );
      };
      var mouseleave = function() {
        element.css( "color", "blue" );
      };
      element.bind('mouseover', mouseover);
      element.bind('mouseleave', mouseleave);

    }
  };
});
```

```
<span ngh-replace-dir style="font-weight: bold; color: blue;" class="spanBackground">
  <strong>Todo esto desaparece</strong><br />
  <em>Todo esto desaparece</em>
</span>
```


Directive restricción E

elementDir

```
.directive('elementDir', function () {  
  return {  
    scope: true,  
    restrict: "E",  
    template: "<strong>elementDir</strong>"  
  };  
})
```

```
<element-dir></element-dir>
```


Directive restricción A

attributeDir: valor por parámetro

```
.directive('attributeDir', function () {  
    return {  
        scope: true,  
        restrict: "A",  
        template: "<strong>attributeDir:</strong> {{value}}",  
        link: function (scope, element, attrs)  
        {  
            scope.value = attrs.attributeDir;  
        }  
    };  
})
```

```
<div attribute-dir="valor por parámetro" />
```


Directive con controlador - aislada

Nombre de item

add

0. Item: uno

```
.directive('withControl', function () {
  return {
    scope: {},
    restrict: 'AE',
    template: '<input placeholder="Nombre de item" class="form-control" type="text" ng-model="nombre" /><br><input type="button" value="add" class="close" ng-click="delete($index)"><span aria-hidden="true">&times;</span></button>{{$index}}. Item: <b>{{nombre}}</b>',
    controller: function($scope, $compile, $http) {
      $scope.list = ["uno", "dos", "tres"]
      $scope.addChild = function() { // this refers to the controller
        $scope.list.push($scope.nombre)
      };
      $scope.delete = function(indx) { // this refers to the controller
        $scope.list.splice(indx, 1)
      };
    }
  };
});
```

```
<div with-control>
```


Directive con atributos: @ & =

Ejecutar

```
.directive('withAttrs', function () {
  return {
    restrict: 'EA',
    scope: {
      datasource: '=',
      add: '=',
      tit: '@titulo'
    },
    controller: function ($scope) {

      $scope.addCustomer = function () {
        $scope.add();
      };
    },
    template: '<button class="btn btn-primary" ng-click="addCustomer()">Ejecutar</button><br><h4>{{tit}}</h4><ul>'
  };
})
```

```
<div with-attrs titulo="Listado" datasource="customers" add="addCustomer"></div>
```


AngularJS Directive

En base a EX_06
construir:

Cras justo odio
Dapibus ac facilisis in
Morbi leo risus
Porta ac consectetur ac
Vestibulum at eros

Param arreglo de
objetos

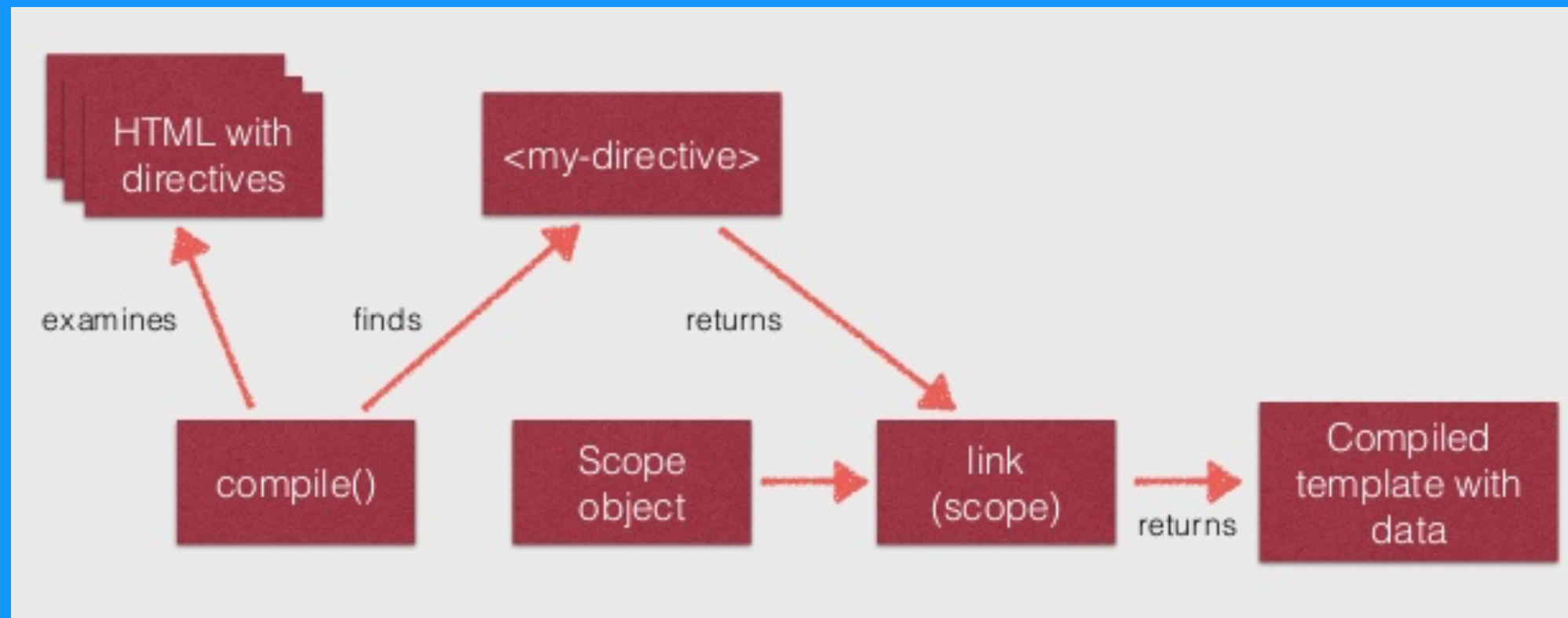
Param true/false para
que sea una lista
“eliminable”

Poder agregar
elementos

AngularJS Directive - Link function

“Tratamiento del DOM, scope y agregamos event listeners”

- ✦ **scope**: El que se pasa a la directiva, al no especificarlo queda el scope del controlador padre.
- ✦ **elem**: Elemento de jQuery, podemos ocupar bind, cambiar css, etc.
- ✦ **attrs**: representa los atributos y valores de nuestro tag.



AngularJS Directive - Link function

“Tratamiento del DOM, scope y agregamos event listeners”

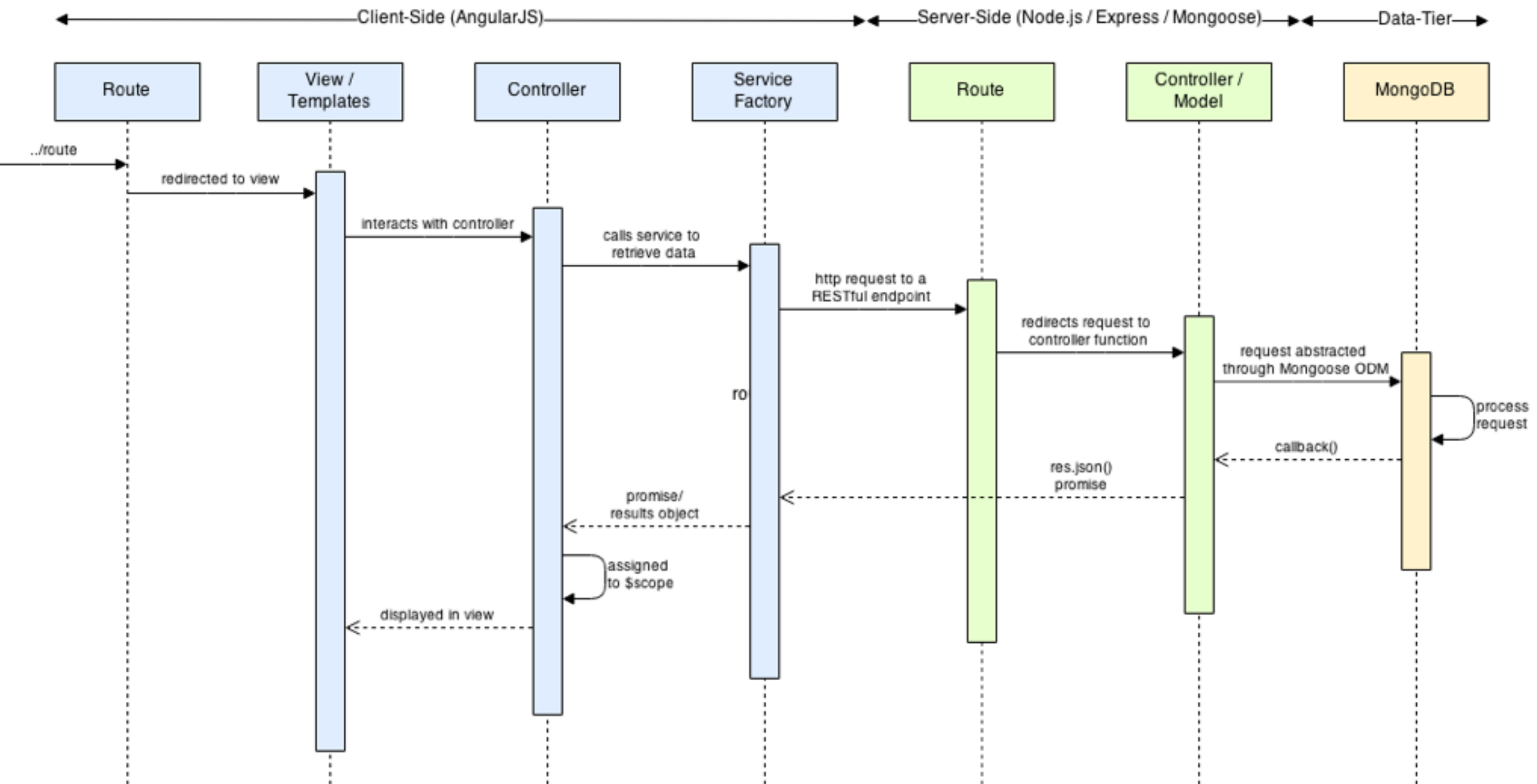
```
angular.module('directivesModule')
.directive('domDirective', function () {
  return {
    restrict: 'A',
    link: function ($scope, element, attrs) {
      element.bind('click', function () {
        element.html('You clicked me!');
      });
      element.bind('mouseenter', function () {
        element.css('background-color', 'yellow');
      });
      element.bind('mouseleave', function () {
        element.css('background-color', 'white');
      });
    }
  };
});
```


AngularJS Forms, ng-show, ng-model, ng-check

EX_07

EX_08

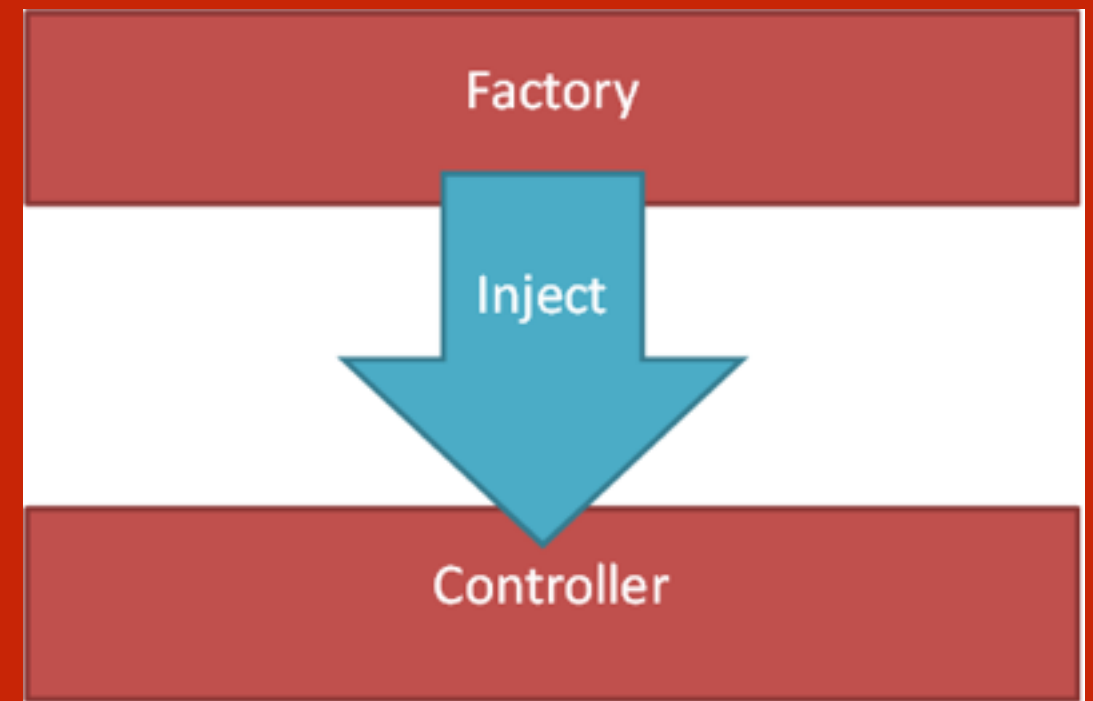
AngularJS MEAN(Mongo, Express, Angular, Node) Flujo Típico



AngularJS Servicios

- **constant** puede ser inyectada donde sea. El valor no cambia
- **value** es un valor simple, inyectable.
- **service** para funcionalidades específicas, conversiones, aprses, etc, inyectable.
- **factory** enfocada en recursos Rest, inyectable. (por lo general es todo lo que necesitamos)

```
app.constant('MOVIE_TITLE', 'Rocky I');  
app.value('movieTitle', 'Rocky II');  
  
app.service('movie', function () {  
    this.title = 'Rocky III';  
});  
  
app.factory('movie', function () {  
    return {  
        title: 'Rocky IV';  
    }  
});
```



EX_09

jsonplaceholder.typicode.com

Routes

All HTTP verbs are supported.
View usage [examples](#).

GET	/posts
GET	/posts/1
GET	/posts/1/comments
GET	/comments?postId=1
GET	/posts?userId=1
POST	/posts
PUT	/posts/1
PATCH	/posts/1
DELETE	/posts/1

EJEMPLOS - TALLER

EX_09 (List)

```
.factory('datosService', function($http) {  
  return {  
    getComments: function(id){  
      return $http.get("http://jsonplaceholder.typicode.com/posts/"+id+"/comments");  
    }  
  }  
});
```

service

```
.controller('DashCtrl', function($scope, datosService) {  
  
  datosService.getComments(1).then(function(response){  
    $scope.listadoc = response.data;  
  
  });  
  
})
```

controller

```
<ul class="list">  
  <a class="item {{$index%2==0?'item-calm':''}}" ng-repeat="it in listadoc" href="#">  
    {{$index+1}} {{it.name}}<p>{{it.body}}</p>  
  </a>  
</ul>
```

html

AngularJS Eventos

mini_pro1

Loading con class=""

Usar css