

Your browser **doesn't support the features required** by impress.js, so you are presented with a simplified version of this presentation.  
For the best experience please use the latest **Chrome, Safari** or **Firefox** browser.



# <Superheld/>

Web-Applikationen mit AngularJS

## Joachim Weinbrenner

jsolutions



## Was ist AngularJS?

- Javascript MV\*-Framework
- by Google
- modular
- lesbar
- erweiterbar
- testbar

# Basics

## AngularJS Ausdruck

```
<html>
  <head>
    <script src="js/angular.js"></script>
  </head>
  <body ng-app>
    <p>Ein einfacher Angular Ausdruck: {{ 1 + 1 }}</p>
  </body>
</html>
```

Ergebnis:

Ein einfacher Angular Ausdruck: 2

## Hello Angular

```
Dein Name: <input type="text" ng-model="name">
<p>Hello {{name}}!</p>
```

Ergebnis:

Dein Name:

Hallo !

## Controller

```
<div ng-controller="ToggleCtrl">
  <button ng-click="toggle()">ON/OFF</button>
  <p ng-show="sichtbar">Hello Controller!</p>
</div>
```

```
function ToggleCtrl($scope) {
  $scope.sichtbar = true;
  $scope.toggle = function() {
    $scope.sichtbar = !$scope.sichtbar;
  };
}
```

ON/OFF Hello Controller!

## Filter

```
Dein Name: <input type="text" ng-model="name">
<p>Hello {{name | uppercase}}!</p>
```

Ergebnis:

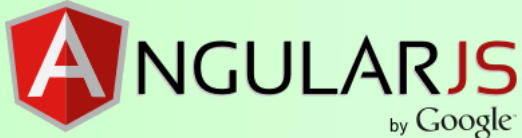
Dein Name:

Hallo !

## Directive

```
<div ng-init="pos = 'Unbekannt'">
  
  <button ng-mouseover="pos = 'Über Button'">
```

```
Ein Button</button>
<p>Position: {{pos}}</p>
</div>
```



Ein Button

Position: Unbekannt

# Controll

## Scope und Wert

```
<div ng-controller="DatumCtrl">
  <p>{{datum}}</p>
</div>
```

```
var DatumCtrl = function($scope) {
  $scope.datum = new Date();
};
```

"2014-03-16T20:27:10.378Z"

# Funktionen im Controller

```
<div ng-controller="SqrtCtrl">
  Eine Zahl: <input type="text" ng-model="zahl"></input>
  Quadrat: {{getSqrt()}}
</div>
```

```
var SqrtCtrl = function($scope) {
  $scope.zahl = 1;

  $scope.getSqrt = function() {
    return $scope.zahl * $scope.zahl;
  };
};
```

Eine Zahl:

Quadrat: 1

# Verschachtelte Controller

```
var app = angular.module("JLApp", []);

app.controller("ElternCtrl", function($scope) {
  $scope.name = "Tom";
});

app.controller("KindCtrl", function($scope) {
  $scope.name = "Fritz";
});
```

```
<body ng-app="JLApp">
  ...
  <div ng-controller="ElternCtrl">
    Eltern-Name: <input type="text" ng-model="name">
    <div class="nested" ng-controller="KindCtrl">
      Kind-Name: <input type="text" ng-model="name">
```

```
Kind-Eltern-Name:  
  <input type="text" ng-model="$parent.name">  
</div>  
</div>
```

Eltern-Name:

Kind-Name:

Kind-Eltern-Name:

# Directive

Zahlreiche mitgelieferte Directives

z.B.:

- ngHide / ngShow
- ngClick
- Mouse-Events (z.B. ngMouseenter)
- Key-Events (z.B. ngKeydown)
- ngRepeat

■ ...

## Beispiel: ngDisabled

```
<label>
  <input type="checkbox" ng-model="checked"/>
  Ich stimme zu!
</label>
<button ng-disabled="!checked">Jetzt bestellen</button>
```

Ich stimme zu!

## Beispiel: ngRepeat

```
app.controller("DevsCtrl", function($scope) {
  $scope.devs = [
    { name: "Lisa", speciality: "HTML/JS" },
    { name: "Kim", speciality: ".Net" },
    { name: "Berta", speciality: "Java" },
    { name: "Xaver", speciality: "AngularJS" }
  ];
});
```

```
<div ng-controller="DevsCtrl">
  <ul>
    <li ng-repeat="dev in devs">
      {{dev.name}} ({{dev.speciality}})
    </li>
  </ul>
</div>
```

- Lisa (HTML/JS)
- Kim (.Net)
- Berta (Java)
- Xaver (AngularJS)

## Eine einfache eigene Directive

```
<body ng-app="JLApp">
...
<label for="checkbox">
  <input id="checkbox" type="checkbox"
    ng-model="sichtbar">ON/OFF
</label>
<div showme="sichtbar">
  <p>ON/OFF per Directive</p>
</div>
```

```
var app = angular.module("JLApp", []);
app.directive("showme", function() {
  return {
    link: function(scope, element, attributes) {
      scope.$watch(attributes.showme, function(value) {
        element.css('display', value ? '' : 'none');
      });
    }
  };
});
```

ON/OFF



# <superheld>-Directive

```
app.directive('superheld', function() {  
  return {  
    restrict: 'E',  
    template:  
      '<div>SUPER-<span ng-transclude></span>!!!</div>',  
    transclude: true  
  };  
});
```

```
<superheld>  
  toll  
</superheld>
```

SUPER- toll !!!

## Directive Options

```
restrict    => E(Element), A(Attribute),  
            C(Class), M(Comment)  
priority    => Auswertereihenfolge  
template, templateUrl => Template  
replace     => Element ersetzen  
transclude => Kindelemente im Template platzieren  
scope      => neuen Scope anlegen  
link       => Dynamisches Data Binding  
compile    => DOM Template beim Laden manipulieren  
...
```

# Filter

## Mitgelieferte Filter

z.B.:

- currency
- date
- json
- upper-/lowercase
- orderBy
- ...

## Beispiel: date

```
<div ng-controller="DatumCtrl">  
  <p>Datum: {{datum | date:"dd.MM.yyyy"}}</p>  
</div>
```

Datum: 16.03.2014

## Beispiel: orderBy

```
<div ng-controller="DevsCtrl" ng-init="reverse='false'">
  <a href="" ng-click="reverse=!reverse">asc/dsc</a>
  <ul>
    <li ng-repeat="dev in devs | orderBy:'name':reverse">
      {{dev.name}} ({{dev.speciality}})
    </li>
  </ul>
</div>
```

### asc/dsc

- Berta (Java)
- Kim (.Net)
- Lisa (HTML/JS)
- Xaver (AngularJS)

## Eigener Filter

```
<div ng-controller="DevsCtrl">
  <ul>
    <li ng-repeat="dev in devs | entferne:'Berta'">
      {{dev.name}} ({{dev.speciality}})
    </li>
  </ul>
</div>
```

```
app.filter("entferne", function() {
  return function(input, entferne) {
```

```
var erg = [];  
for (var i=0; i<input.length; i++) {  
    if (input[i].name !== entferne) {  
        erg.push(input[i]);  
    }  
}  
return erg;  
};  
});
```

- Lisa (HTML/JS)
- Kim (.Net)
- Xaver (AngularJS)

## Verkettung von Filtern

```
<div ng-controller="DevsCtrl" ng-init="reverse='false'">  
  <a href="" ng-click="reverse=!reverse">asc/dsc</a>  
  <ul>  
    <li ng-repeat="dev in devs | entferne:'Berta'  
              | orderBy:'name':reverse">  
      {{dev.name}} ({{dev.speciality}})  
    </li>  
  </ul>  
</div>
```

asc/dsc

- Kim (.Net)
- Lisa (HTML/JS)
- Xaver (AngularJS)

# Services

Zahlreiche mitgelieferte Services, z.B.:

- \$http
- \$locale
- \$log
- \$rootScope
- \$timeout
- \$window

Beispiel: Ein Countdown

```
function TimerCtrl($scope, $timeout) {  
  $scope.timer = 10;  
  
  $scope.start = function() {
```

```
var stop = $timeout(function() {
  if($scope.timer > 0){
    $scope.timer = $scope.timer - 1;
    $scope.start();
  } else {
    $timeout.cancel(stop);
  }
}, 1000);
};
```

```
<div ng-controller="TimerCtrl">
  <button ng-click="start()">START</button>
  <p>Noch {{timer}} sec!</p>
</div>
```

START

Noch 10 sec!

## Ein einfacher Beispielservice:

```
app.factory("OSService", function() {
  var os = [ "Linux", "MacOS", "Windows" ];
  return {
    all : function() {
      return os;
    },
    first : function() {
      return os[0];
    }
  };
});
```

```
app.controller("OSAllCtrl", function($scope, OSService) {
    $scope.oss = OSService.all();
});

app.controller("OSFirstCtrl", function($scope, OSService) {
    $scope.firstOS = OSService.first();
});
```

```
<div ng-controller="OSAllCtrl">
  <ul>
    <li ng-repeat="os in oss">{{os}}</li>
    <li ng-controller="OSFirstCtrl"><i>First: {{firstOS}}</i></li>
  </ul>
</div>
```

- Linux
- MacOS
- Windows
- *First: Linux*

# Praxis

## AJAX-Requests

```
<div ng-controller="BuecherCtrl">
  <ul>
    <li ng-repeat="buch in buecher">{{buch.titel}}</li>
  </ul>
</div>
```

```
app.controller("BuecherCtrl", function($scope, $http) {
  $http.get('exampledata/buecher.json').
  success(function(data, status, headers, config) {
    $scope.buecher = data;
  }).
  error(function(data, status, headers, config) {
    // log error
  });
});
```

- AngularJS
- Recipes with AngularJS
- Javascript: The Good Parts



- Javascript Pocket Reference

## Methoden in \$http

- \$http.get
- \$http.head
- \$http.post
- \$http.put
- \$http.delete
- \$http.jsonp

## RESTful: ngResource

```
app.factory('Buecher', ['$resource', function($resource) {  
  return $resource('/buecher/:buchId',  
    {buchId: '@id'},  
    {kaufen: {method: 'POST',  
              params: {kaufen: true},  
              isArray: false}});  
}]);
```

<i>Call</i>	<i>Method</i>	<i>URL</i>	<i>Return</i>
<code>Buecher.query()</code>	GET	<code>/buecher</code>	JSON Array
<code>Buecher.get({id:</code>	GET	<code>/buecher/47</code>	Single

47}))			JSON
<code>Buecher.save({}, buch)</code>	POST	/buecher mit post data "buch"	Single JSON
<code>Buecher.save({id: 48}, buch)</code>	POST	/buecher/48 mit post data "buch"	Single JSON
<code>Buecher.remove({id: 47})</code>	DELETE	/buecher/47	Single JSON
<code>Buecher.delete({id: 47})</code>	DELETE	/buecher/47	Single JSON

## Buecher im Einsatz

```
// alle Bücher laden:
var buecher = Buecher.query();

// Ein Buch laden und direkt im Callback arbeiten:
Buecher.get({id: 123}, function(buch) {
  buch.autor = "Bugs Bunny";
  // non-GET Methoden werden auf die Instanzen gemapped:
  buch.$save();
  // auch unsere Custom-Funktion:
  buch.$kaufen({anzahl:2});
  // Erzeugt POST: /buecher/123?anzahl=2&kaufen=true
  // mit POST Data "buch"
});
```

```
// neues Buch anlegen:
var buch = new Buecher();
buch.autor = "Speedy Gonzales";
buch.titel = "Fiesta Fiasco";
buch.$save();
// Erzeugt POST: /buecher mit
// {autor:'Speedy Gonzales', titel:'Fiesta Fiasco'}
```

# URLs / Routing / Partials

```
var app = angular.module("JLApp", [ngRoute]).
  config(function($routeProvider, $locationProvider) {
    $locationProvider.hashPrefix('!');
    $routeProvider.
      when("/buecher",
        { templateUrl: "partials/buecherlist.html" }).
      when("/buecher/:id",
        { templateUrl: "partials/buchdetails.html",
          controller: "ShowCtrl" }).
      otherwise( { redirectTo: "/buecher" });
  });
```

```
// in index.html: Rahmen mit Menü, Footer, ...
// ...und Partial anzeigen:
<div ng-view></div>
```

```
// Beispiel f. partials/buchdetails.html:
<h3>{{buch.titel}} Details</h3>
<p>Name: {{buch.titel}}</p>
<p>Autor: {{buch.autor}}</p>
<a href="#!buecher">Zurück zur Liste</a>
```

# Testen mit Jasmine

```
describe('SomeCtrl', function() {
  var scope, ctrl;

  beforeEach(inject(function($injector,
```

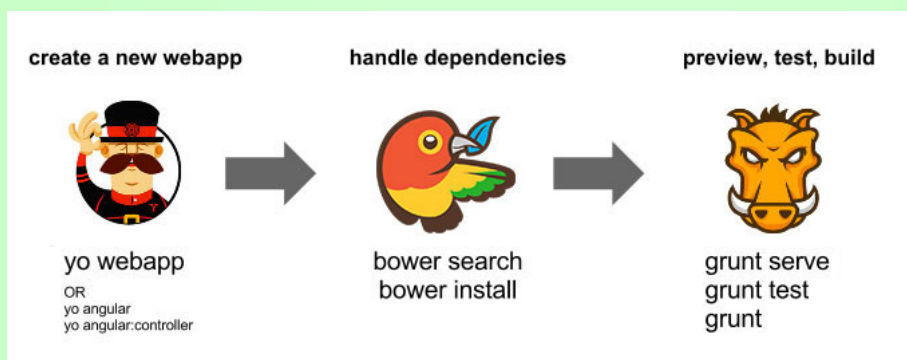
```

        $controller, $rootScope) {
    scope = $rootScope.$new();
    ctrl = $controller(SomeCtrl, { $scope: scope });
    });

    it('should change message if name changed', function() {
        scope.name = "Fritz";
        scope.$digest();
        expect(scope.greeting).toBe("Hello Fritz");
    });
});

```

## Generate / Manage / Build / Test / Deploy: Yeoman



Kurzanleitung (Blog)

## Wo anfangen?

- angular-seed  
<https://github.com/angular/angular-seed>
- angular-sprout  
<https://github.com/jwanga/angular-sprout>
- <https://github.com/cgross/generator-cg->

angular

## Links

- <http://angularjs.org/>
- <http://angularjs.de/>
- <https://egghead.io/>

## Literatur

- AngularJS | Brad Green | O'Reilly
- Recipes with Angular.js | Frederik Dietz | Leanpub



**Vielen Dank!**

Noch Fragen?

Blog: <http://joachim.weinbrenner.name>

Twitter: @weinbrenner

