ng-start

Begin working with AngularJS

From : Vijay Shivakumar

ANGULARJS by Google

Before we Begin

- IDE :
  - Aptana Studio (from www.aptana.com)
- Browsers
  - Chrome
  - Firefox (firebug)
- Utilities
  - Internet Connection

About Me

Vijay Shivakumar

Designer | Developer | Trainer

Adobe | CERTIFIED EXPERT Flex® with AIR

Training on web technologies and Adobe products from past 10+ years

About You

Developing

Testing

UI Designing

Managing

Some Prerequisites

Development Experience with

HTML / XML

CSS

JavaScript

jQuery

JSON

Disclaimer

- I am not here to promote any framework, library or an IDE
- What ever I teach may change in future
- The concepts explained here are my opinion and
  I reserve my right to change my opinion as I upgrade & experience ☺…
- With updates from AngularJS community we may expect up gradations or omissions in future releases.
- We are not going to cover each and every thing about AngularJS in this workshop, you will have to explore several features on your own too…

Why do we need this ?

Trending demands

Utilize client side resources

Faster applications

Scalable yet lean

Maintainable

Well structured

Testable

Considering SPA / SPI

What is SPA / SPI ?

- Single Page Application (SPA) or
  Single Page Interface (SPI) is a static or dynamic web application that fits on a single web page without refreshing the page when the user interacts.

ANGULARJS by Google

SPA Advantage / Disadvantage

Advantages

- Easily converted to RWD
- No page refresh between queries
- Faster and better response
- User Friendly
- Saves Bandwidth

Disadvantages

- Lack of support on older browsers
- Bulky if not organized properly and becomes slow
- Not suitable for every type of application
- Fairly new concept hence shortage of resources and tools

Challenges of SPA

DOM Manipulation

Data Handling

Data Binding

Validation

Formatting

History management

AJAX

and more...

What do I choose ?

ember

batman.js

canjs

dōjō

jQuery write less, do more. zepto.js

Important considerations

- Community / Support
- HTML5 and future compatibility
- REST
- Mobile
- Performance
- Page Speed

AngularJS dazzles above all

- Community / Support
- HTML5 and future compatibility
- REST
- Mobile
- Performance
- Page Speed

History

- Started by Miško Hevery in 2009, while working at Google
- A JavaScript framework that focuses on enhancing browser-based applications with MVC capability.

ANGULARJS by Google

Who is developing it ?

Built and developed by Google

They are paid employees of Google

1000s of contributors and growing on github

What's it for me

Free

Open source

Scalable

Maintainable

Robust

What can AngularJS do ?

Angular JS makes HTML suitable for web application development.

Angular JS is especially used to make SPA.

AngularJS is declarative and has preset ways of working

Current Browser support

IE9 or greater only

Current Versions of

Safari

Chrome

Firefox

Opera

Mobile Browsers

Android

Chrome Mobile

iOS Safari

What is AngularJS

Its MV * opinionated framework

Model : Store data and state of your app

View : Is what user see and interacts

Controller / Presenter / View Model : which stores all the logic of your program

Features of Angular JS

Model View Controller architecture

- A well known and proven architecture

Data Binding

- Automatically synchronizes values between Model and View

Templates

- Makes it very easy to update the user interface

Dependency injections

- Code dependencies are automatically injected where needed

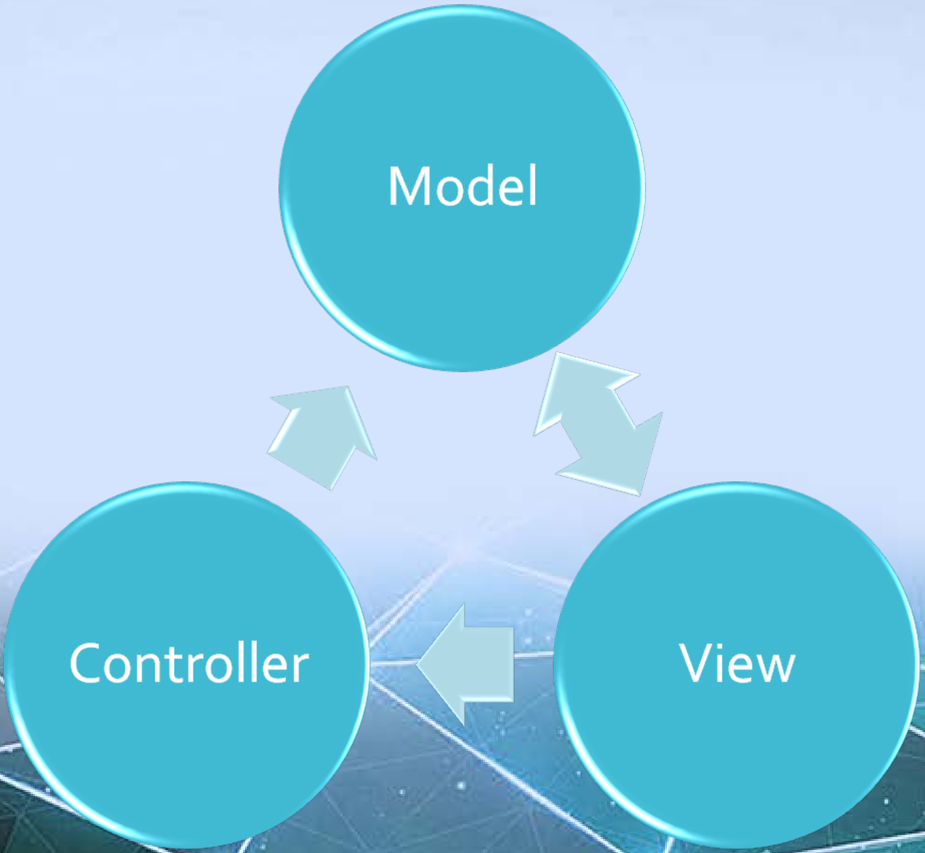
Extends HTML with directives

- Lots of powerful standard directives or create your own

Build with testing in mind

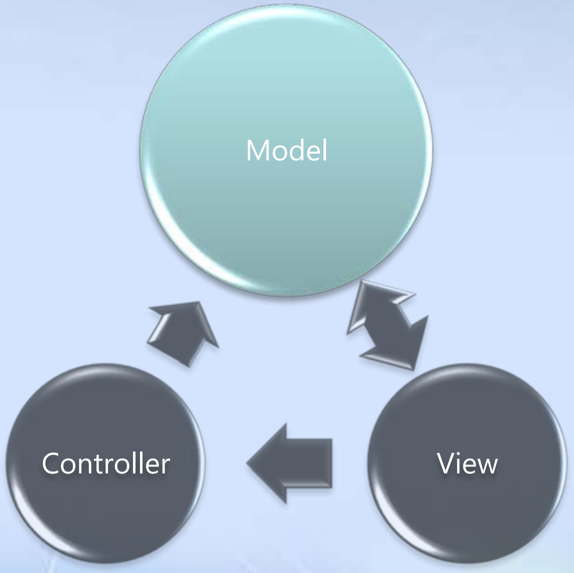
- Makes it much easier to unit test different parts

MVC in Angular



ANGULARJS by Google

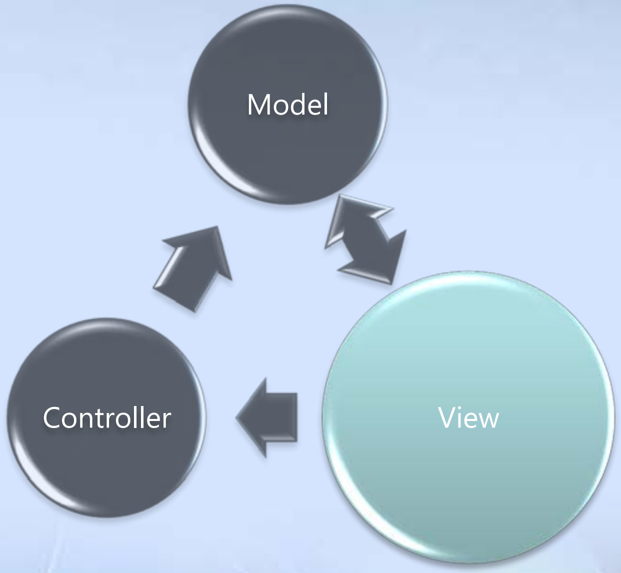
The Model



The business data

Exposed to the view through the $scope

The View



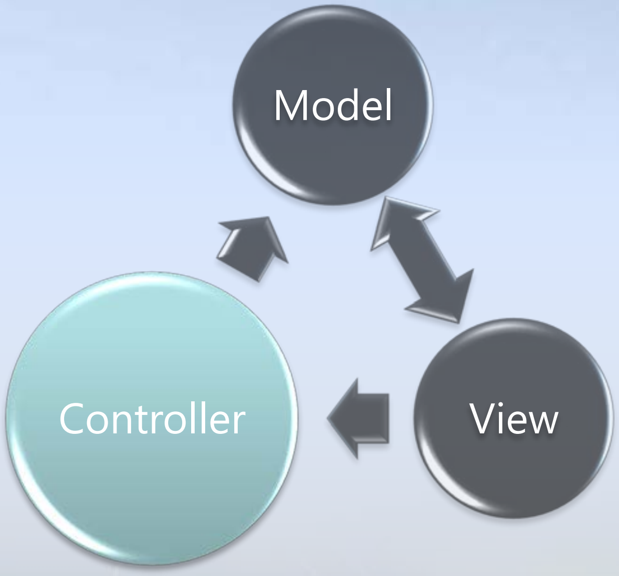
The user interface layer

Data binds to the model

Calls functions on the controller

Use declarative directives for reusable code

The Controller

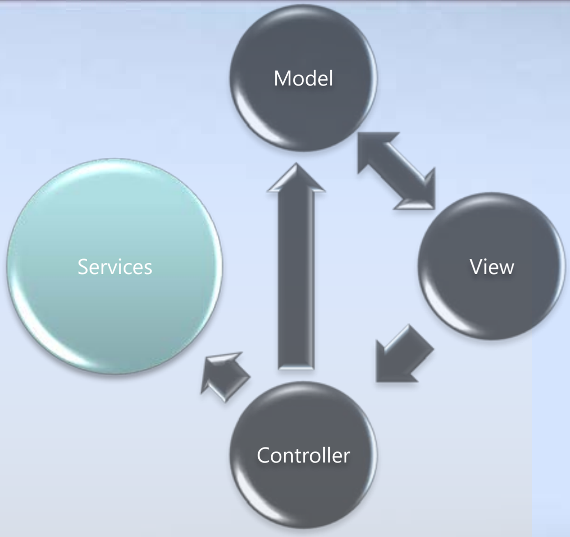


Glues the view and the model together

Provides additional functionality

Uses additional services for reusable logic

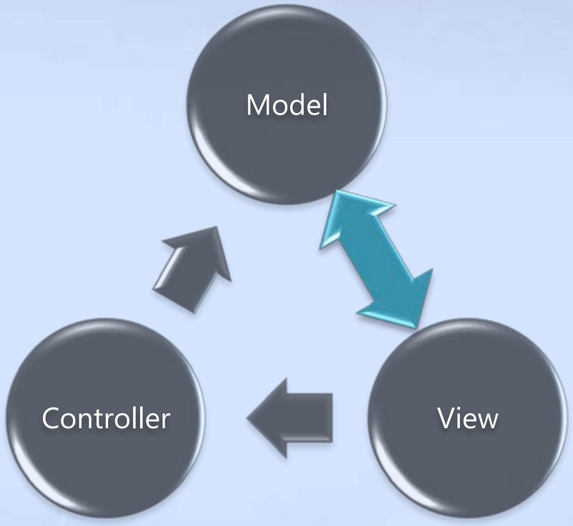
The Services



- Services are reusable pieces of business logic
  - Separation results in reuse and testability
- Created as singleton objects
  - Inject in to controllers by AngularJS using dependency injection
- Services are created as part of a module
  - One module can take a dependency on another module
- Modules are groupings of related functionality
  - Also used to bootstrap the application

ANGULARJS by Google

The Scope Chain



$scope ?

The glue between the Model and View

What will we learn ?

- Expressions
- Data Binding
- Views
- Directives
- Filters
- Controllers
- Scope
- Modules
- Routing
- Services
- Factories
- Animation

This is how it starts

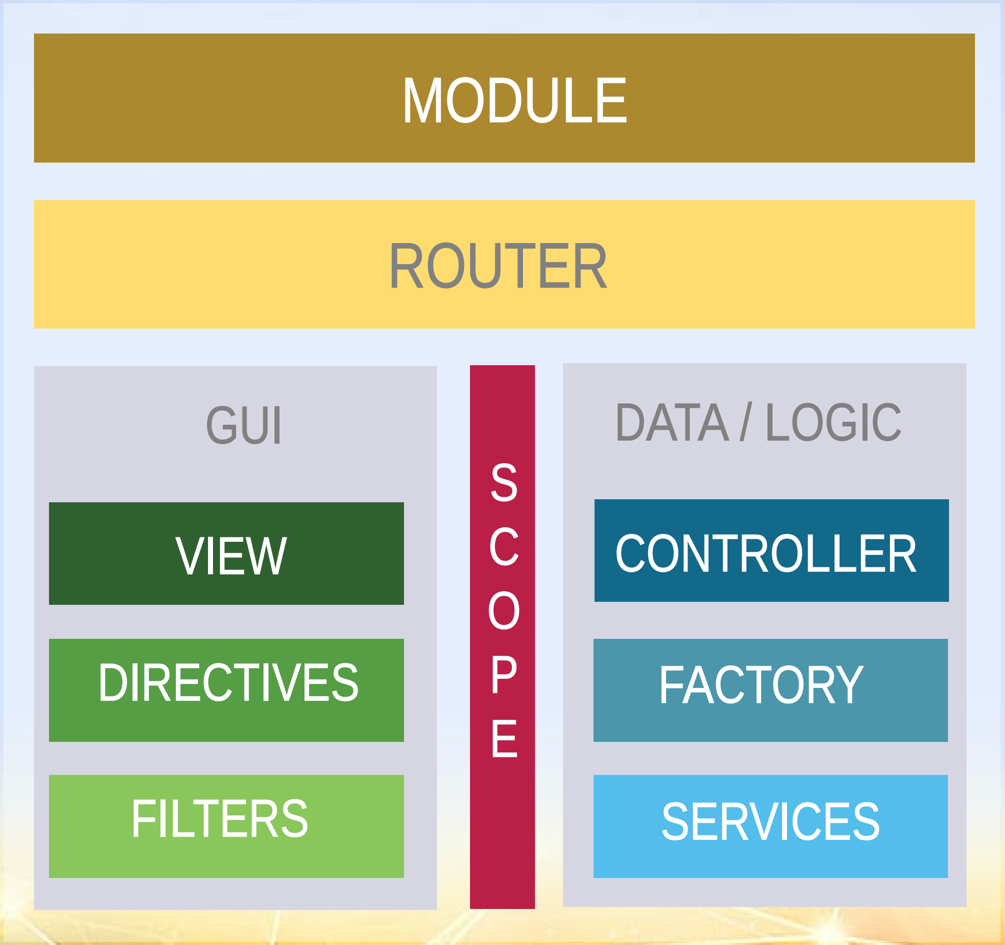
Get AngularJS in your HTML page

Add ng-app directive

Start using AngularJS

ANGULARJS by Google

Anatomy of AngularJS



What is a Module

MODULE

It's a container for

Controllers , Routers , Factories, Services

Directives, Filters, Animation

ANGULARJS by Google

What is a Router

ROUTER

It’s a location for View

Router decides which **view** and what **controller** gets loaded for the user

What is a View

VIEW

What the user sees on the screen

Used to display the data to the user

User can interact with the application using the view

Events are created by this component

What are Directives

DIRECTIVES

Teach HTML new tricks

Generates the html tag to be displayed

Eg., button, drop down, text input or even some custom tags

What are Filters

FILTERS

Filters apply formatting to the data

Show them in upper case, lower case, format them

What are Controllers

CONTROLLERS

Can hold static data , handle user events etc

What are Factories

FACTORY

Connect to external resource for data

What are Services

SERVICES

Connect to external resource for data

Expressions

Expressions in AngularJS is useful to apply some simple logic in the view.

They are evaluated in double curly braces {{}}

Simple Math : 1+2, 2*3, 9/3,etc.

String Concatenation : “Hello ”+”World”

Access Object Properties : user.name

Access Array values : items[index]

Data Binding

Binding data from model to views

Supports 2 way data binding

* New in 1.3X supports one time binding

Binding can be done with double curly braces {{prop}} or with ng-bind directive

Filters

- filter
- currency
- date
- orderBy
- uppercase
- limitTo
- lowercase
- number
- json

Built-in RPC

Built in Recipes for RPC

Provider Recipe

Factory Recipe

Service Recipe

Constant Recipe

Value Recipe

Factories & Services

About Factories and Services

AngularJS includes several built-in factories and services

They are a singleton

They are used to do some repetitive tasks by controllers(ajax, business logic, share data between controllers, etc.,)

They are similar in feature and differ by implementation.

Both are inherited from a provider

Factory

What are Factories ?

Why we create custom factories

- Handle repetitive tasks
- Share data between controllers

How to create them

- Create with module.factory() similar to controller
- A regular function that returns a custom object
- Can be injected to other components
- Can have dependencies

Service

What are Services ?

Why we create custom services

- Handle repetitive tasks
- Share data between controllers

How to create them

- Create with module.service()
- The function that you create will be the service object
- Can be injected to other components
- Can have dependencies

AJAX in AngularJS

ANGULARJS by Google

Few Built-in Factories & Services

$http	: make xmlhttprequests
$timeout	: similar to setTimeout
$window	: AngularJS way of accessing window
$location	: To access navigator.location
$q	: Used by $http and returns promise
$interval	: similar to setInterval
$filter	: used to factory custom filters
$log	: used to logging
$resource	: used for RESTApis

ANGULARJS by Google

Ajax

$http : Communicates with the remote HTTP servers via browser's XMLHttpRequest object or via JSONP

.get, .head, .post, .put, .delete, .jsonp, .patch

$resource : Creates a more refined object to deal with RESTful server-side data sources

$resource(url, [paramDefaults], [actions], options);

DOM Events

DOM Events

ng-click	ng-mousedown	ng-keydown
ng-dblclick	ng-mouseup	ng-keyup
ng-change	ng-mouseenter	ng-keypress
	ng-mouseleave	
	ng-mousemove	
	ng-mouseover	

ANGULARJS by Google

Thank you

vijay.shivu@gmail.com