#### Introductie

- Wat is je ervaring in web development?
- Wat voor applicaties bouw je?
- Is er iets specifieks wat je hoopt te leren?





## Introductie



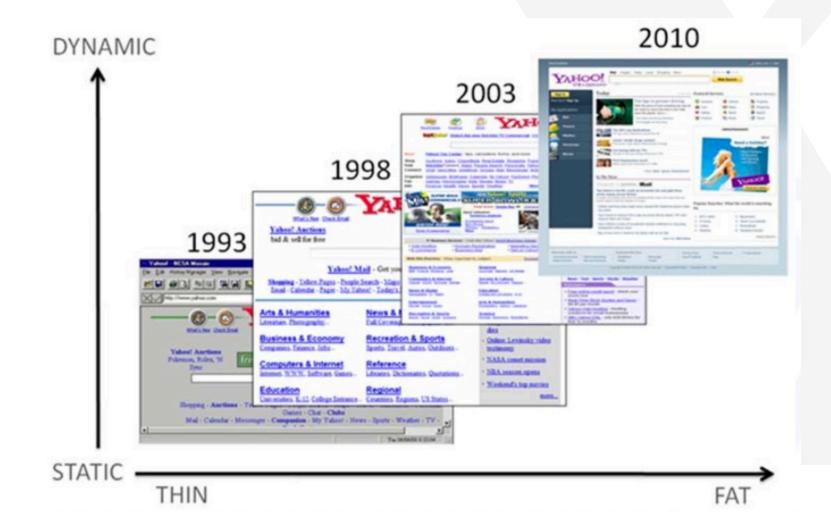


- Why
- What
- Next 3 days



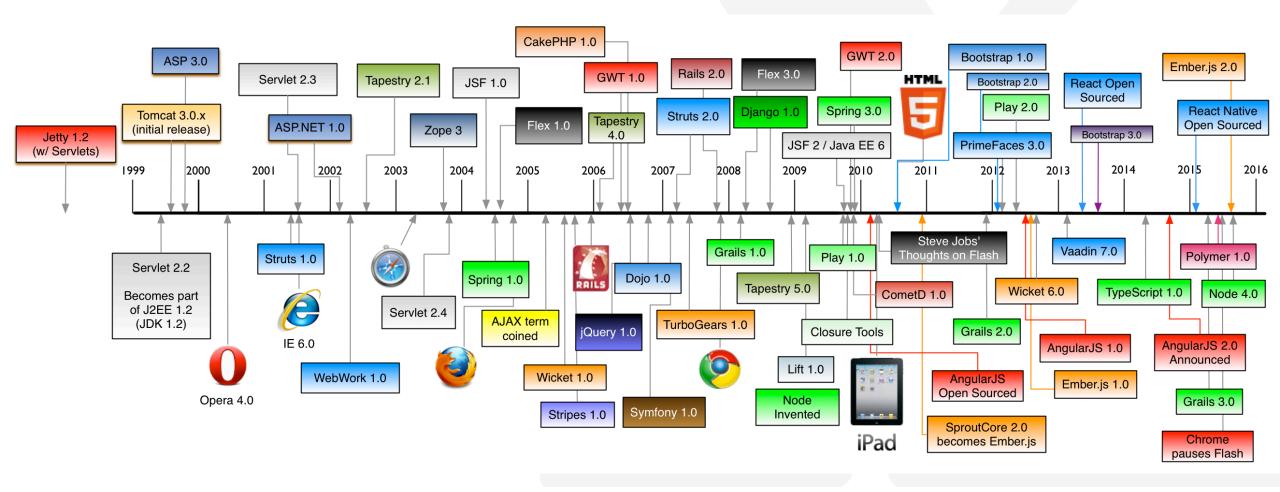


#### Web applications evolution





#### Web technologies history





#### Current technologies - "Rich client"

User Interaction

JavaScript + Web APIs

Presentation

HTML 5 & CSS 3

Transport

HTTP, HTTPS, SPDY (WebSocket, WebRTC, RESTful..)

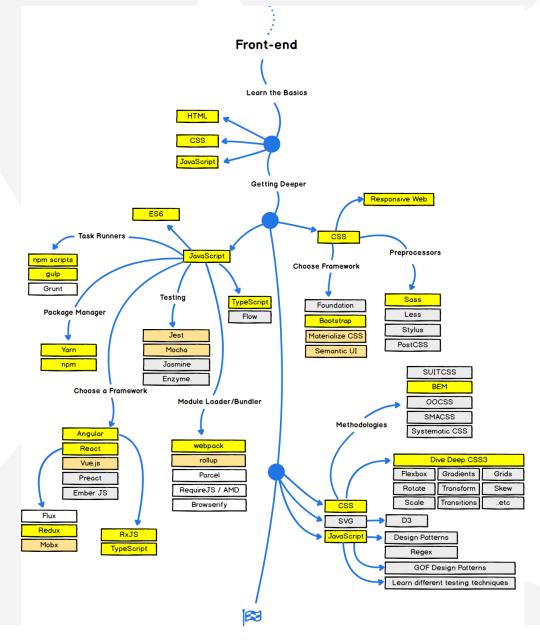
Resources

Web Resource URIs (Service, Content, Site)



#### Current technologies - problems

- Fast moving environment
- Code maintainability
- Many solutions for a problem





- Easy to use
- Fast
- Easily maintainable
- "Go with the flow"





### **Angular ideals**

Stay up to Date

major version every 6 months

**TypeScript** 

RxJS

Performance

highly optimized code from Angular compiler

Internal rewrites (drop-in upgrades)

**Mobile first** 

code splitting

full responsive support Testable

Integrated test tools (karma, protractor, jasmine)

Framework Test Helpers



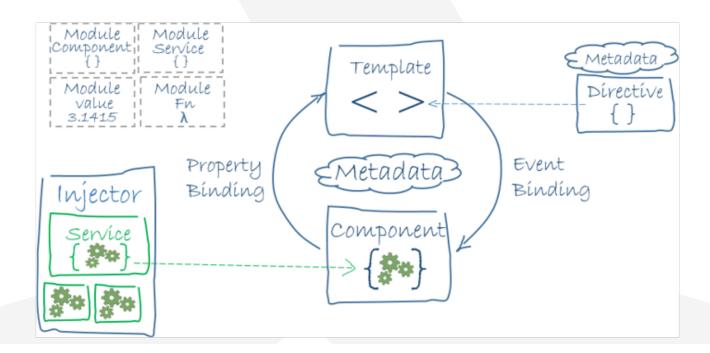
- Why
- What
- Next 3 days





#### Angular - what it does

- Application architecture
- Template compilation
- Runtime engine
- Dependency Injection





#### Angular - what it has

Schematics CLI Forms Routing **IDE Support** Bazel HTTP Security CDK i18n PWA Elements Universal & more! Animation Angular Material







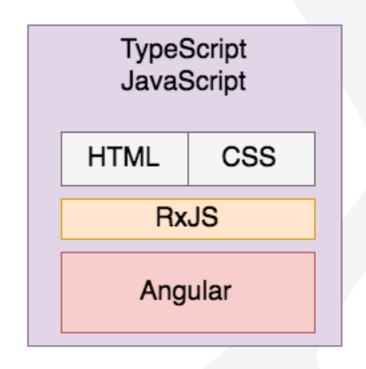




TypeScript
JavaScript

Node npm



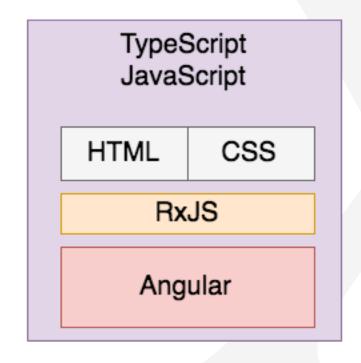


Node npm



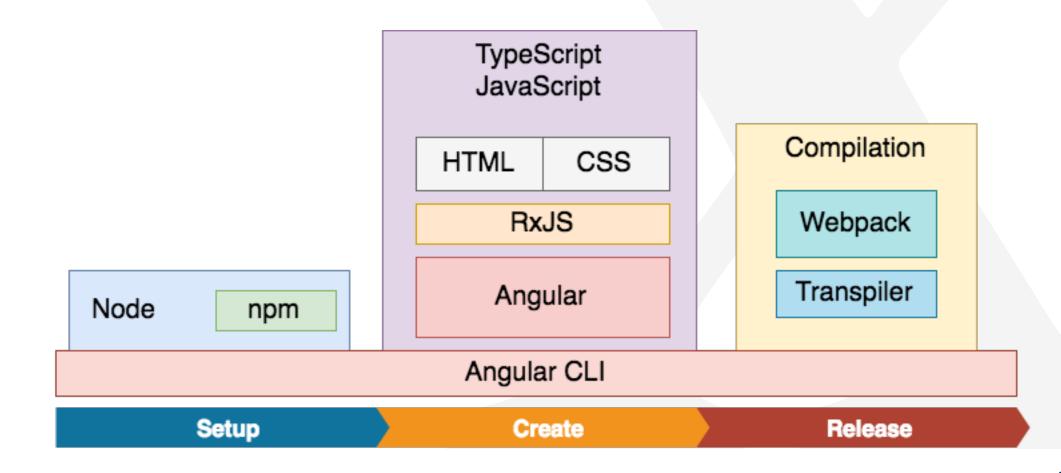
npm

Node



Compilation
Webpack
Transpiler







- Why
- What
- Next 3 days





#### Day 1

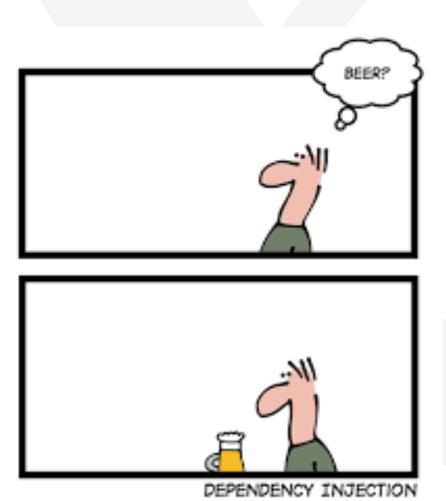
- Introduction Angular
- Modules & Components
- Template Syntax





## Day 2

- Dependency Injection
- Testing
- Forms





## Day 3

- Routing
- RxJS
- Library / CLI



## **During each day**

- Theory
- Code walkthrough
- Application: BookShop



