

Taking an Angular App to Production



Joe Eames

WEB DEVELOPER

@josepheames www.joeeames.me



Agenda

Linting

Tuning RxJS Requests

Enabling Production Mode

Basic Deployment

AOT Compiler

Tree Shaking & Bundling

Webpack

Optimistic Bundle Download



Preparation Levels

Minimal

Tune RxJS Requests
Enable Prod Mode

Full

Use AOT
Bundle
Minify
Tree Shake

Lazy Loading

Use Lazy Loaded Modules



How Many Modules Should I Create?



Minification, Bundling, Code Splitting, and Tree Shaking



Minification

Source

```
export class orderService {  
  addOrder(order) {  
    ...  
  }  
  
  findOrder(id) {  
    ...  
  }  
}
```



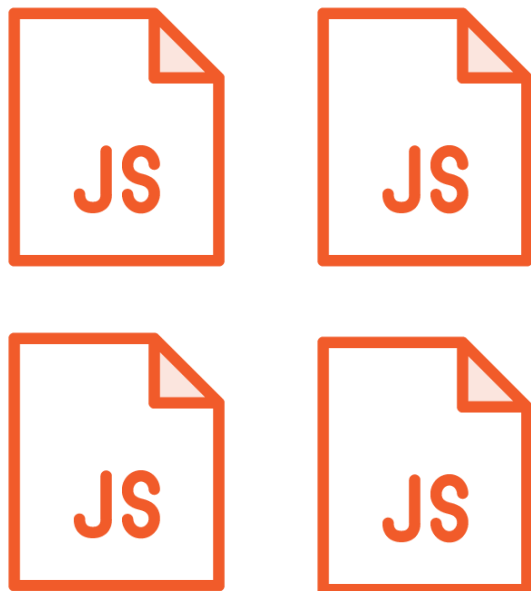
Service

```
export class orderService{add  
Order(order){...}findOrder(id){...}}
```



Bundling

Development

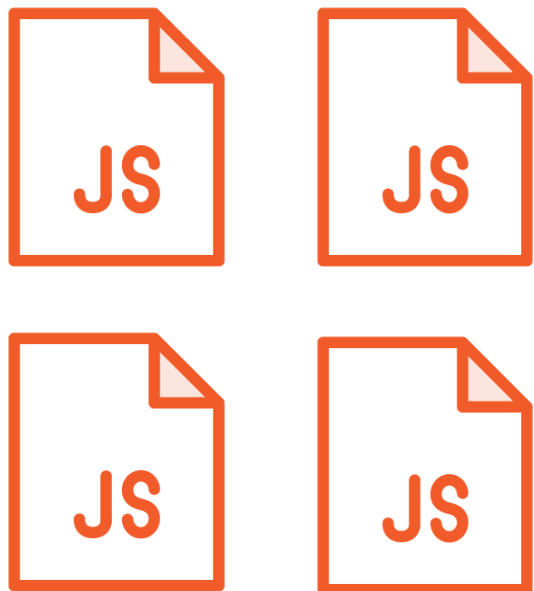


Production



Code Splitting

Bundled



Split



Tree Shaking

Calling Code

```
...  
userService.getUsers()  
...
```



Service

```
export class UserService {  
  getUsers() {  
    ...  
  }  
  
  getUser(id) {  
    ...  
  }  
}
```



Bundler Options

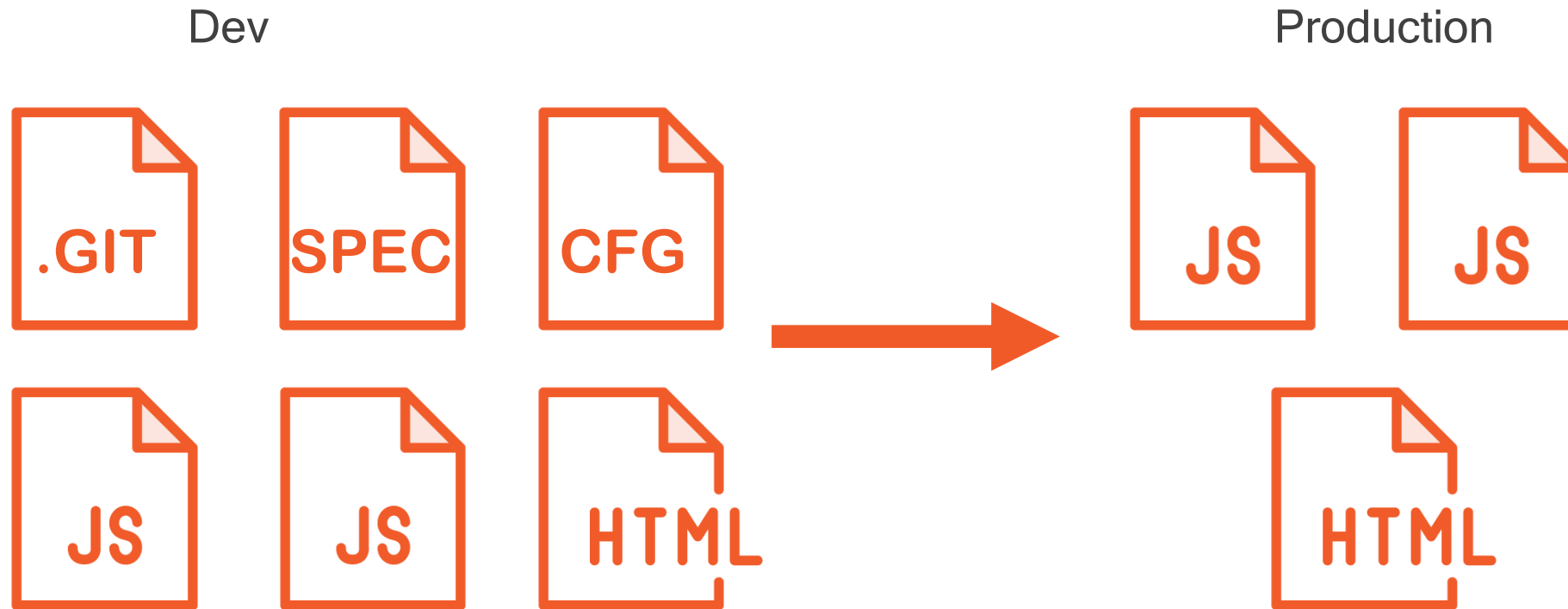
SystemJS

Rollup

Webpack



Simple Copy Deployment



AOT Benefits

Faster Rendering

Fewer Requests

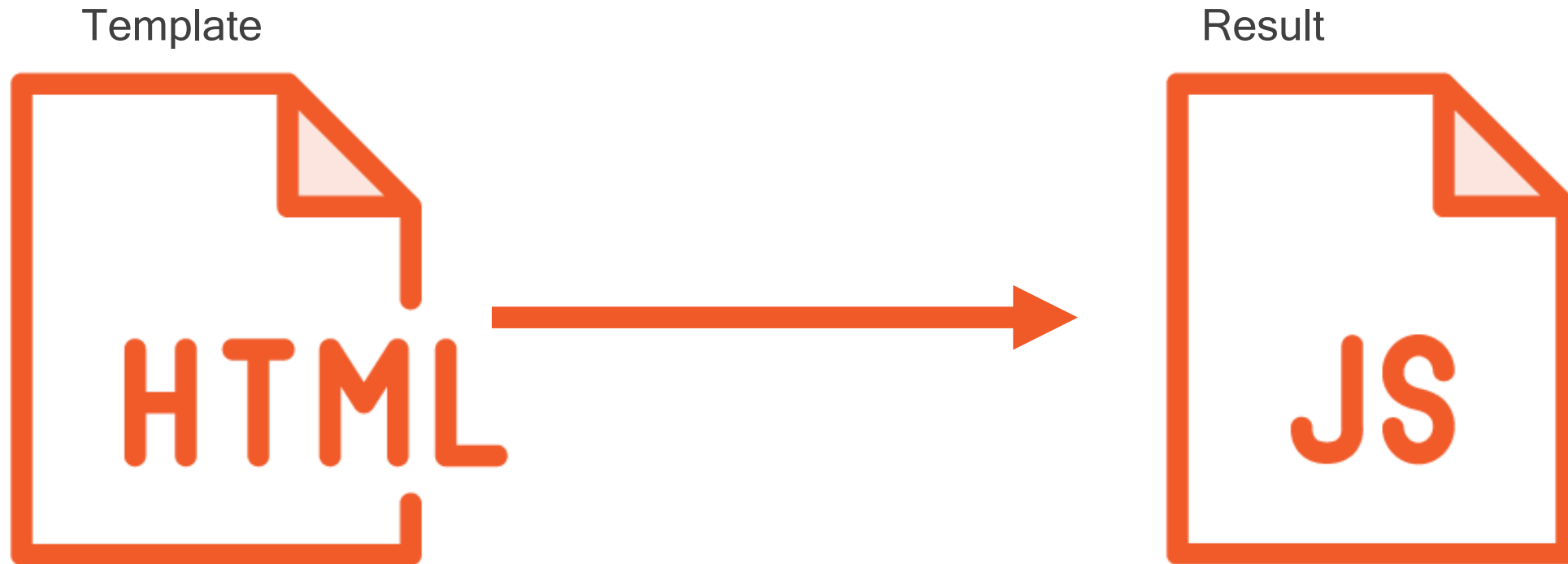
Smaller Angular Framework Download

Detect Template Errors

Better Security



Compiling Templates



```
<div well-title>
  {{session.name}}
  <i *ngIf="session.voters.length > 3"
    class="glyphicon glyphicon-fire"
    style="color:red"> </i>
</div>
```

Development = Just In Time Compiler

Much of Angular Needed in Browser is the Compiler

Takes Time to Compile Templates

Production = Ahead of Time Compiler



AOT No-No's

`form.controls.controlName`

`control.errors?.someError`

Default exports

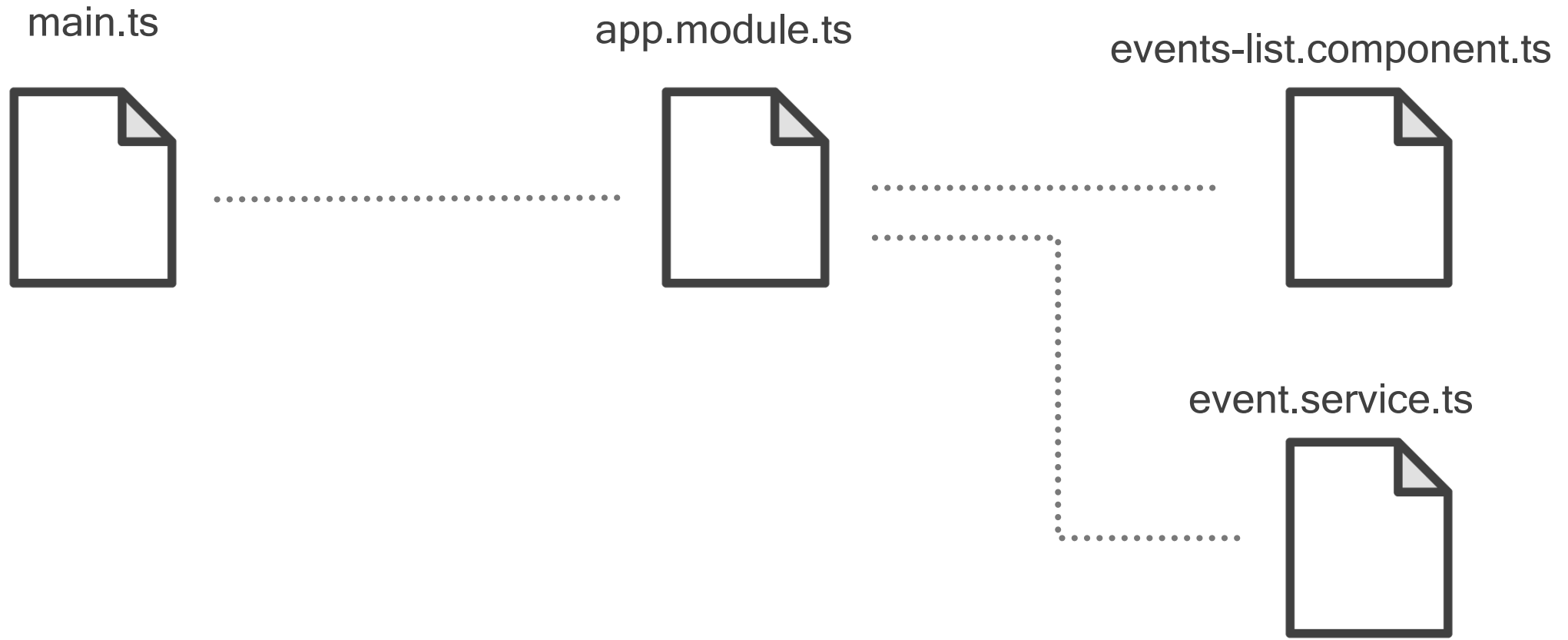
Functions in providers, routes or declarations of a module

Any field used in a template, including Inputs, must be public

declare var for globals



ES6 Module Tree



Summary



Linting

RxJS Requests

Prod Mode

AOT

Bundling & Tree Shaking

Webpack

Optimistic Bundle Download



Thank You

