

Webpack & Neutrino

Jan.schulz@devugees.org

Agenda

1. Introduction
2. Server Side Templating VS. Single Page App
3. How Webpack works
4. Webpack in Action
5. Webpack Loaders

Why?

We will use Webpack & Neutrino for all of our web projects.

1. Introduction

- Webpack is a Build-Tool.

What are Build-Tools?

1. Introduction

- Build-Tools
 1. **Bundling**
 2. Transpile SASS & LESS to CSS
 3. Minify and Uglify
 4. Livereload
 5. Fix JS Errors
 6. Start Scripts (npm run etc.)
 7. Solve Incompatibilities

1. Introduction

- Neutrino is a Scaffolding-Tool.

What are Scaffolding-Tools?

1. Introduction



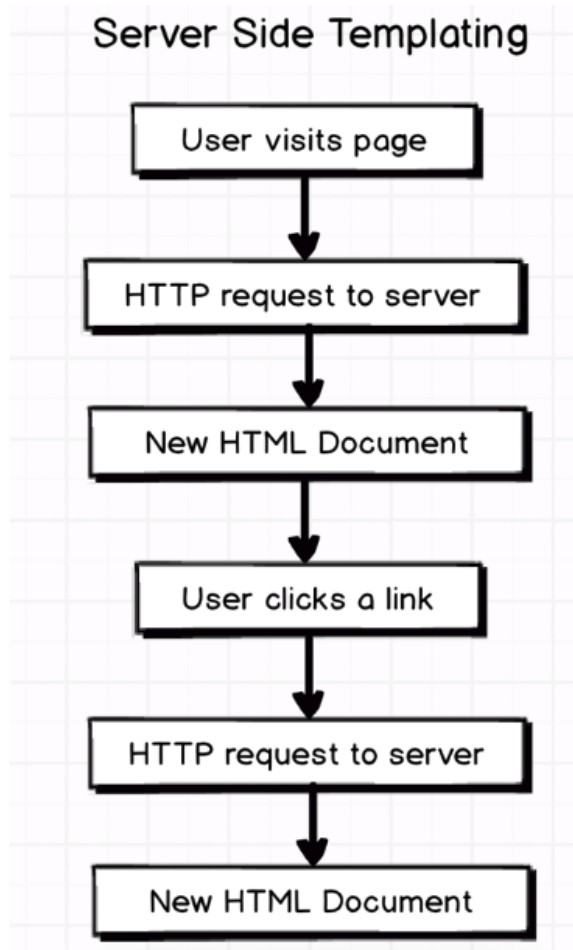
1. Introduction

- Scaffolding:
 1. Building an app with a **shared preset** of configuration.
 2. Other people may build their apps with the same preset as you

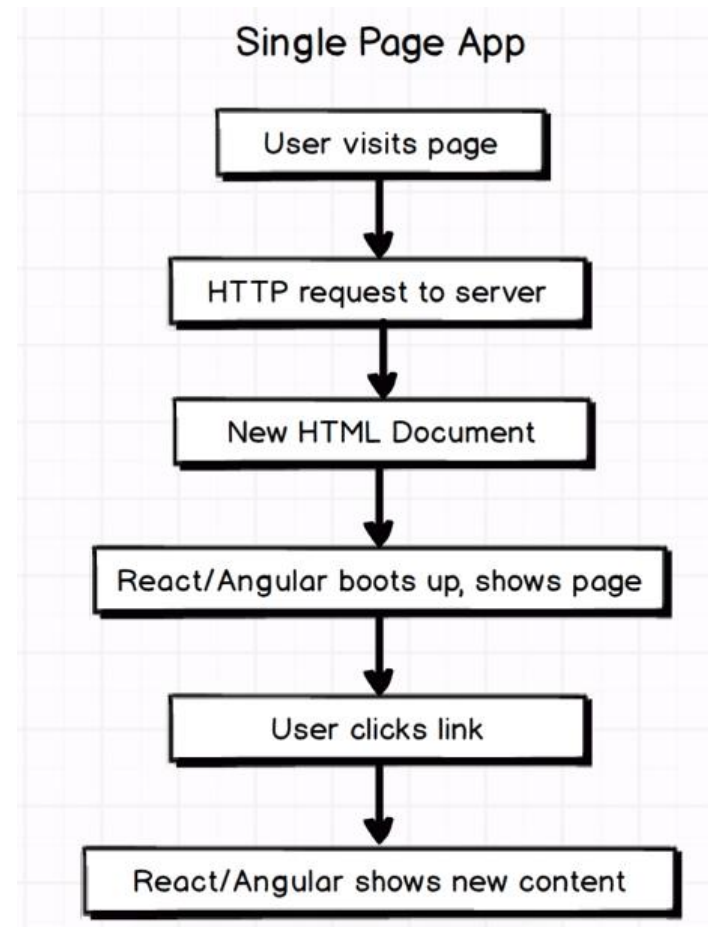
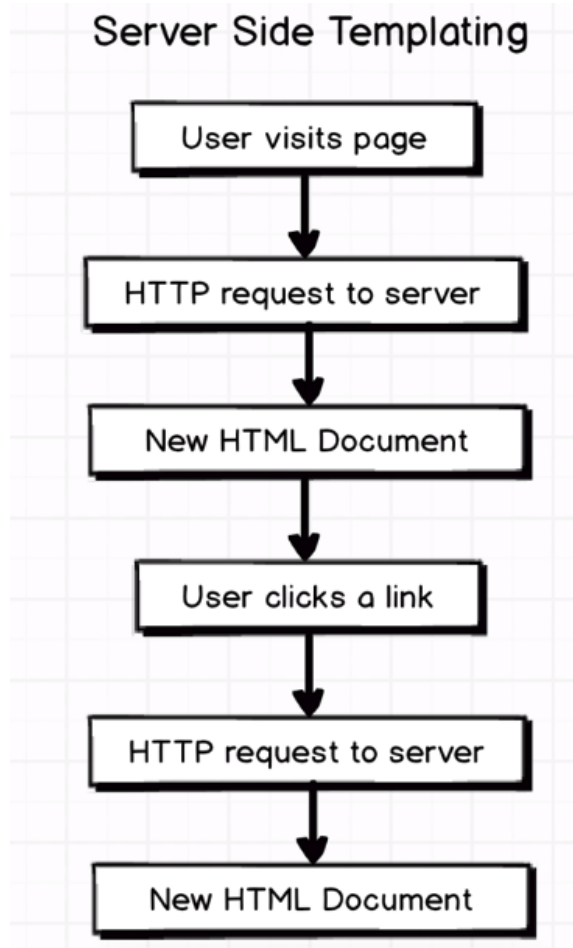
2. Server Side Templating VS. Single Page App

Server Side Templating	Back end server creates an HTML document and sends it to the user
Single Page App	Server sends a bare-bones HTML doc to the user. Javascript runs on the users machine to assemble a full webpage

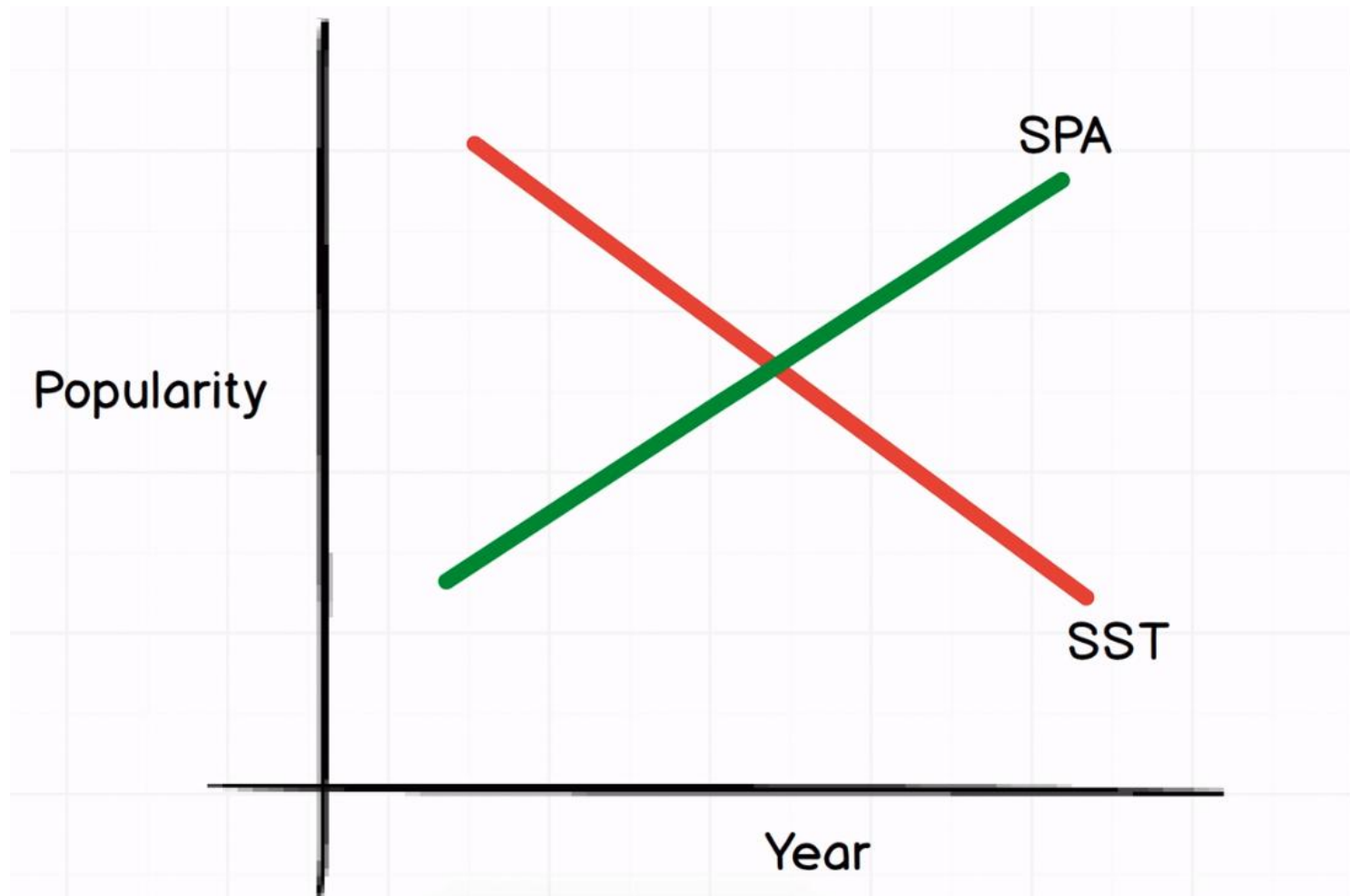
2. Server Side Templating VS. Single Page App



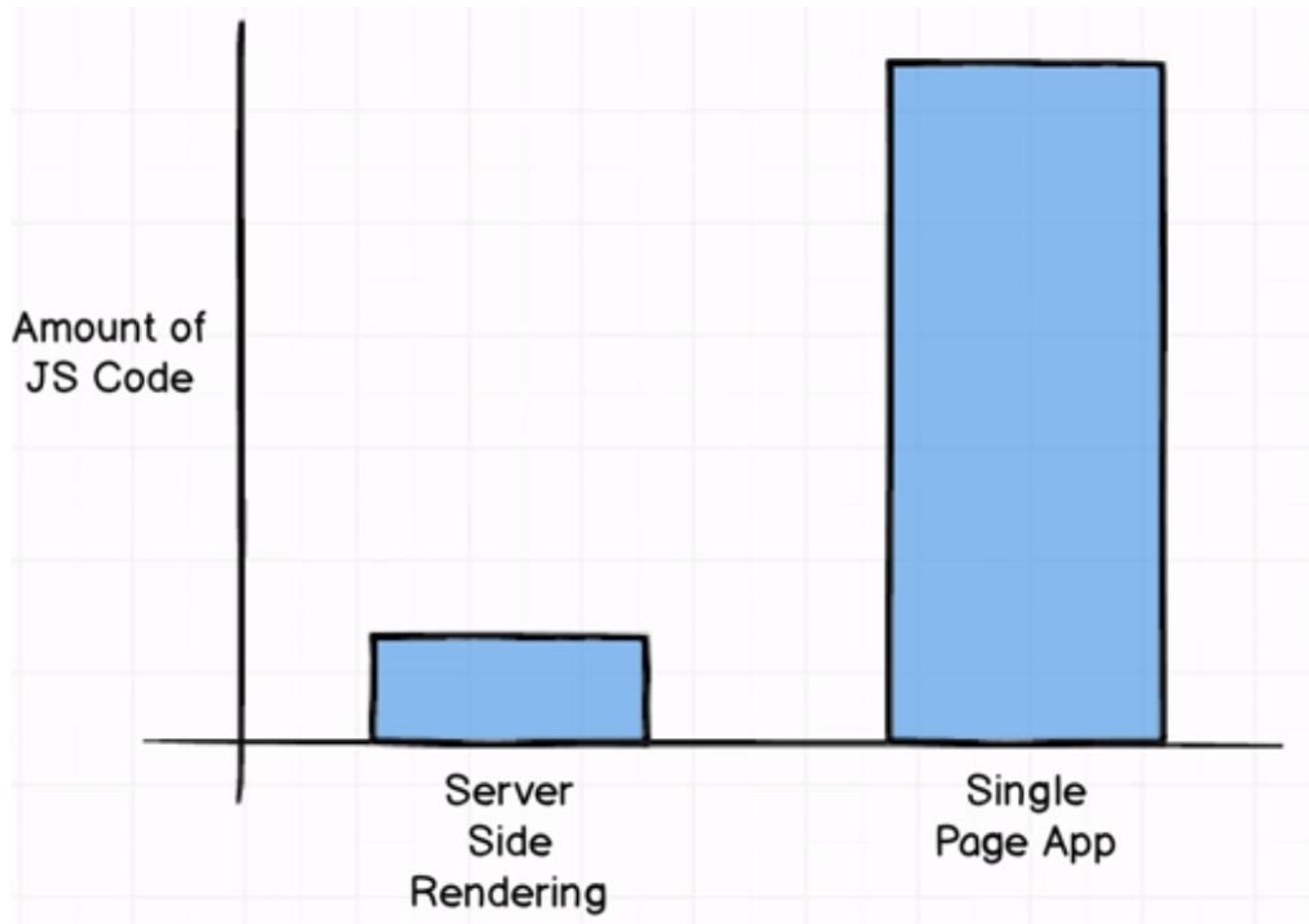
2. Server Side Templating VS. Single Page App



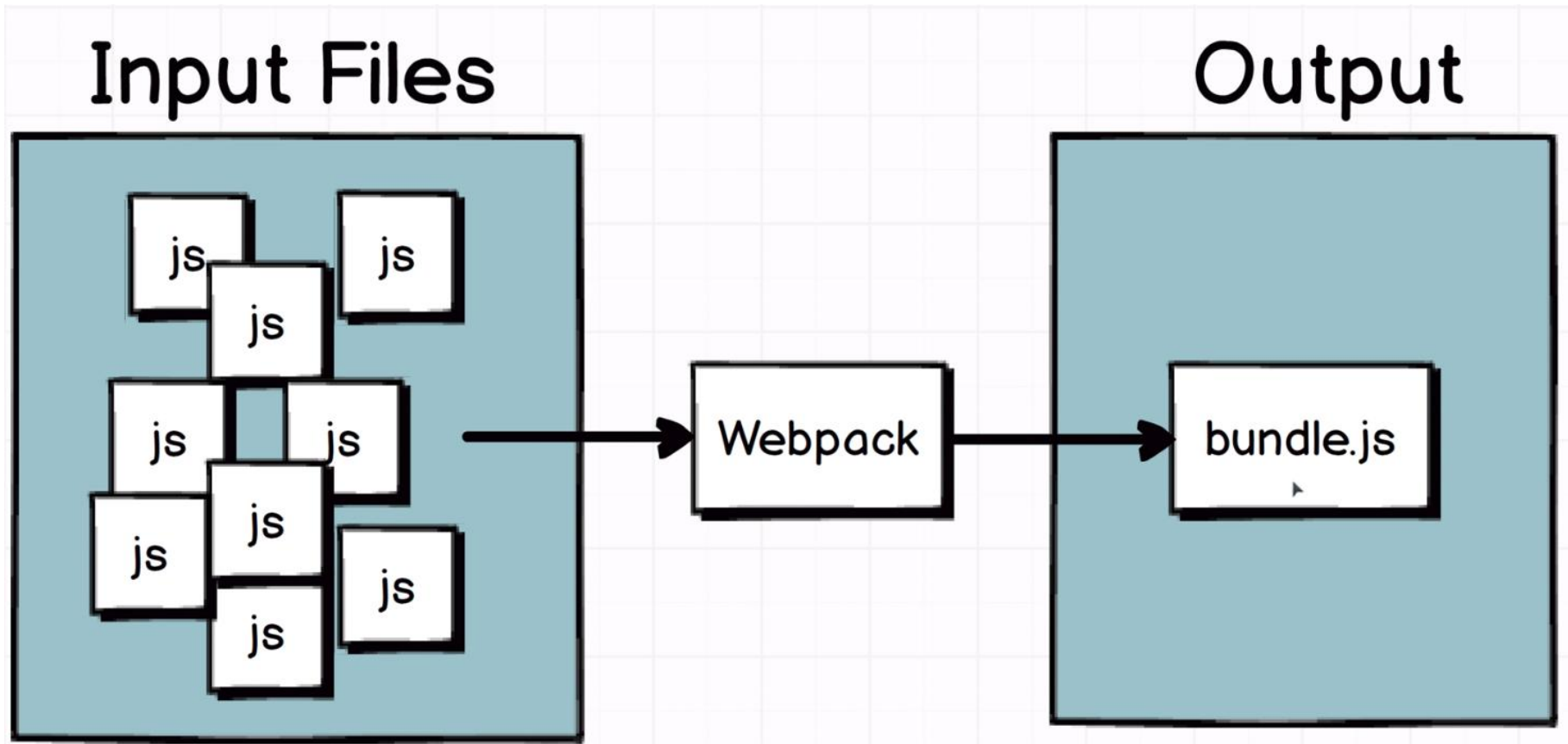
2. Server Side Templating VS. Single Page App



2. Server Side Templating VS. Single Page App



3. How Webpack works



3. How Webpack works

- Modules can be imported/exported in different syntaxes:

Module System	Common Syntax
CommonJS	<div>require</div> <div>module.exports</div>
AMD	<div>define</div> <div>require</div>
ES2015	<div>import</div> <div>export</div>

3. How Webpack works

- Webpack is given **ONE** module and works through all modules, that are related to that module by imports.

3. How Webpack works

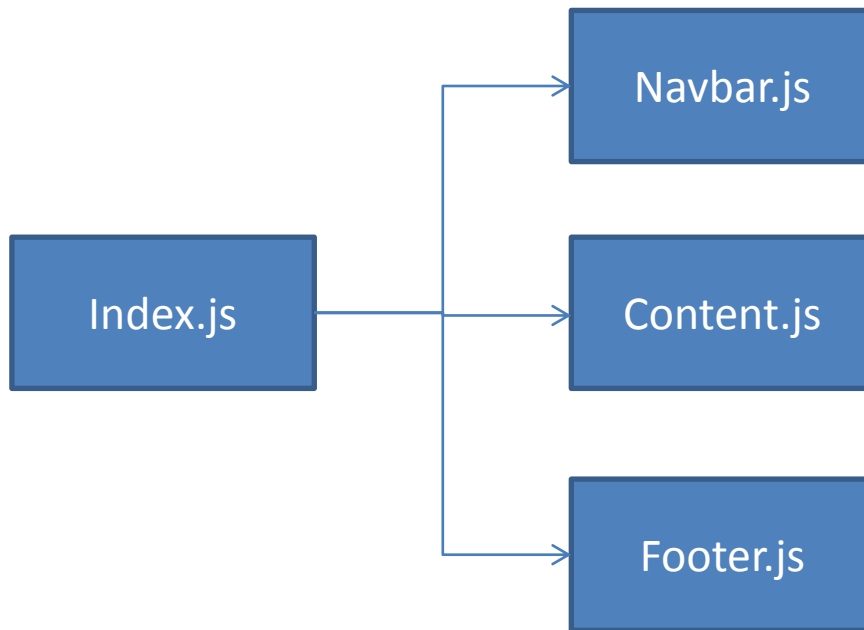
- Webpack is given **ONE** module and works through all modules, that are related to that module by imports.



Index.js

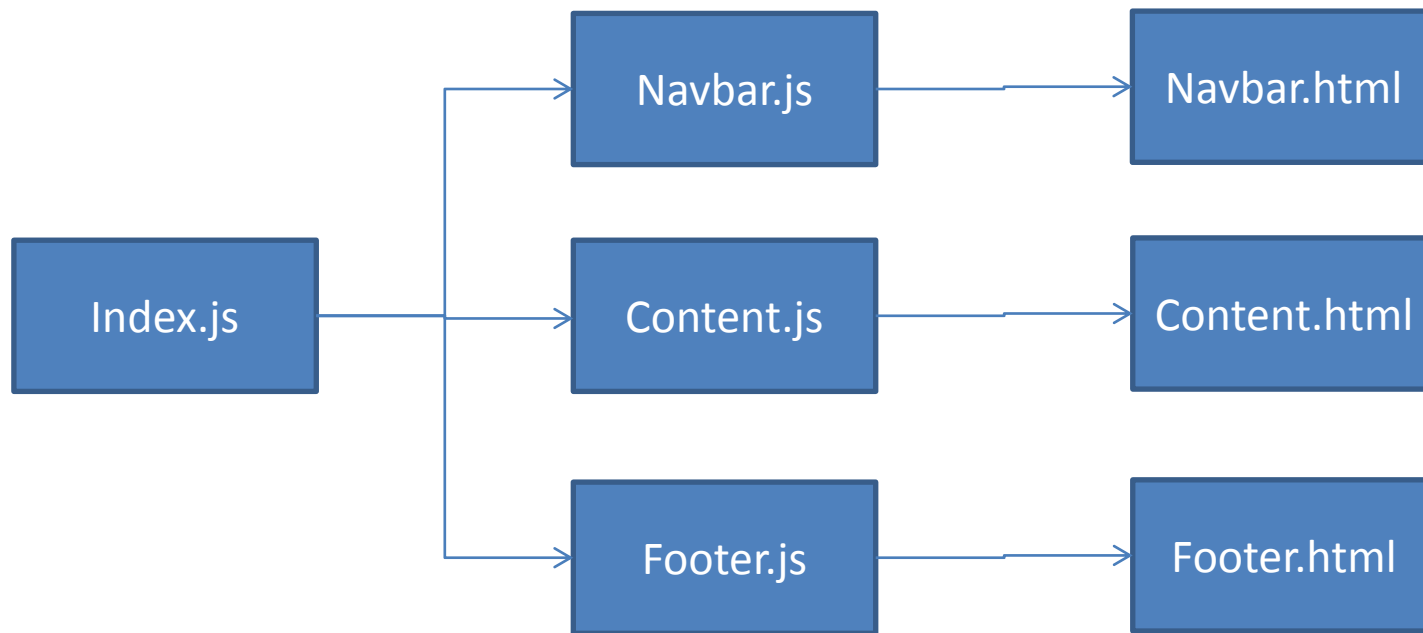
3. How Webpack works

- Webpack is given **ONE** module and works through all modules, that are related to that module by imports.



3. How Webpack works

- Webpack is given **ONE** module and works through all modules, that are related to that module by imports.



5. Webpack Loaders

- Loaders include certain file-types into the transpilation process
 - JS – files
 - HTML – files
 - Pictures (JPG, GIF, PNG)
 - CSS/SASS/LESS files

5. Webpack Loaders



The diagram consists of two blue rectangular boxes with thin dark blue borders. The left box contains the word 'BABEL' in white, uppercase letters. The right box contains the word 'Webpack' in white, title case letters. Below each box is a line of black text describing its function.

BABEL

Turns ES6 to ES5

Webpack

Link Up JS modules
together

5. Webpack Loaders

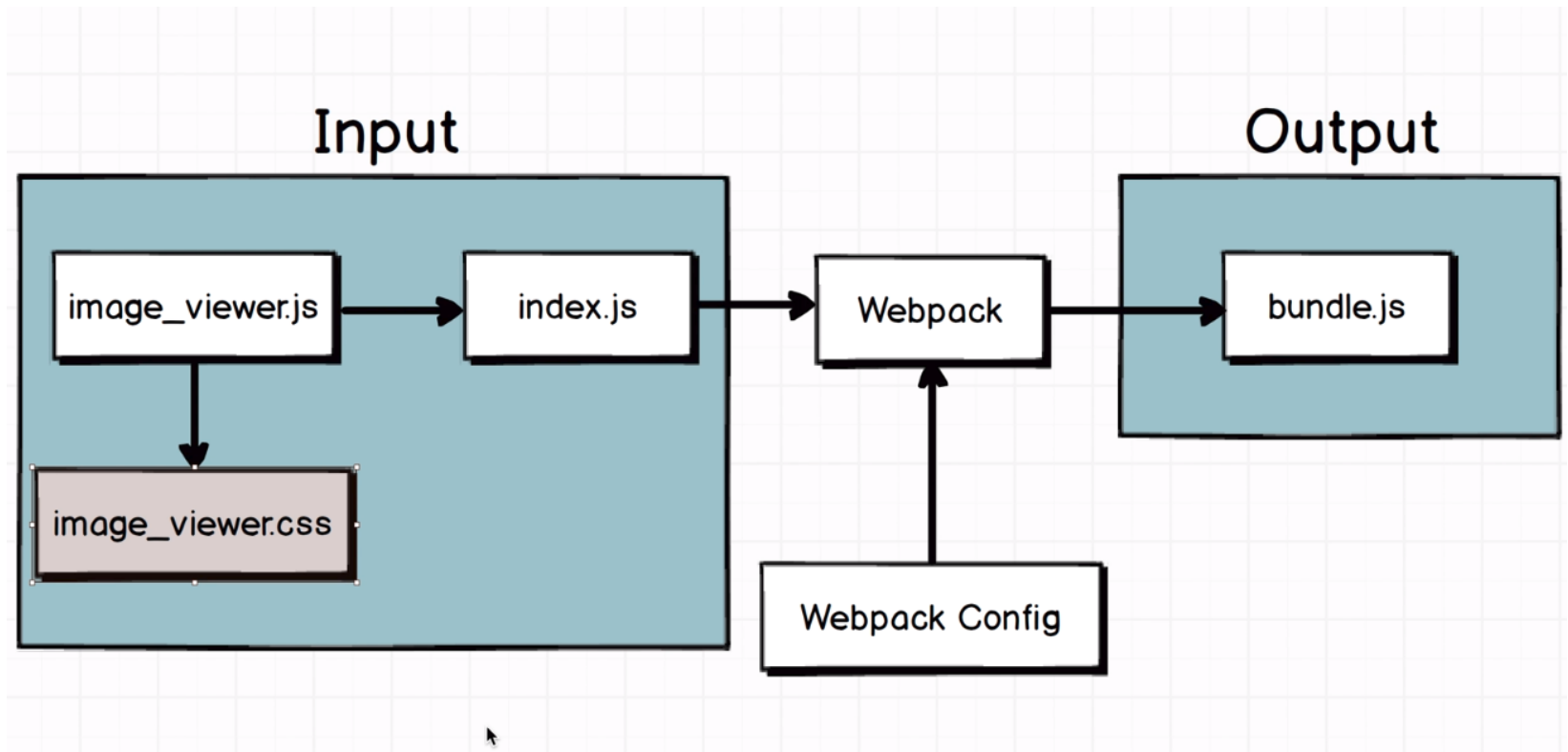
- Babel: Transpiles ES6 to ES5

Module	Purpose
<code>babel-loader</code>	Teaches babel how to work with webpack
<code>babel-core</code>	Knows how to take in code, parse it, and generate some output files
<code>babel-preset-env</code>	Ruleset for telling babel exactly what pieces of ES2015/6/7 syntax to look for, and how to turn it into ES5 code

5. Webpack Loaders

Action	CommonJS	ES2015
Import a module	<pre>const sum = require('./sum');</pre>	<pre>import sum from './sum';</pre>
Export some code	<pre>module.exports = sum;</pre>	<pre>export default sum;</pre>

5. Webpack Loaders



5. Webpack Loaders

css-loader

Knows how to deal
with CSS imports

style-loader

Takes CSS imports and
adds them to the HTML
document

7. Task

Create a mailclient web application fullstack, i.e. the front and the backend using Webpack, JQuery, Bootstrap, NodeJS and Nodemailer.

1. Given are three input boxes. One for the recipient mail address, the second for the subject and the third for the mail body. If the user hits the „Send“ button, an email will be sent to the mail recipient address.
2. When the email is sent, create a new entry in the collection „mails“ on your Mongo database „mailclient“.
2. If everything went okay, show a success message. Otherwise, show an error message.