



Deep Dive Into Webpack


Md Islam (@tanjir)

Senior Software Developer, Klipfolio Inc

<https://www.linkedin.com/in/tanjir/>

What is webpack?



- 
- Module bundler (not only web... name is deceiving)
 - Can process js, css, sass, images and many more!
 - Everything is a module
 - Can transpile codes written in typescript, coffeescript
 - Can pre and post process contents using loaders and plugins
 - Can process files live as you make changes

```

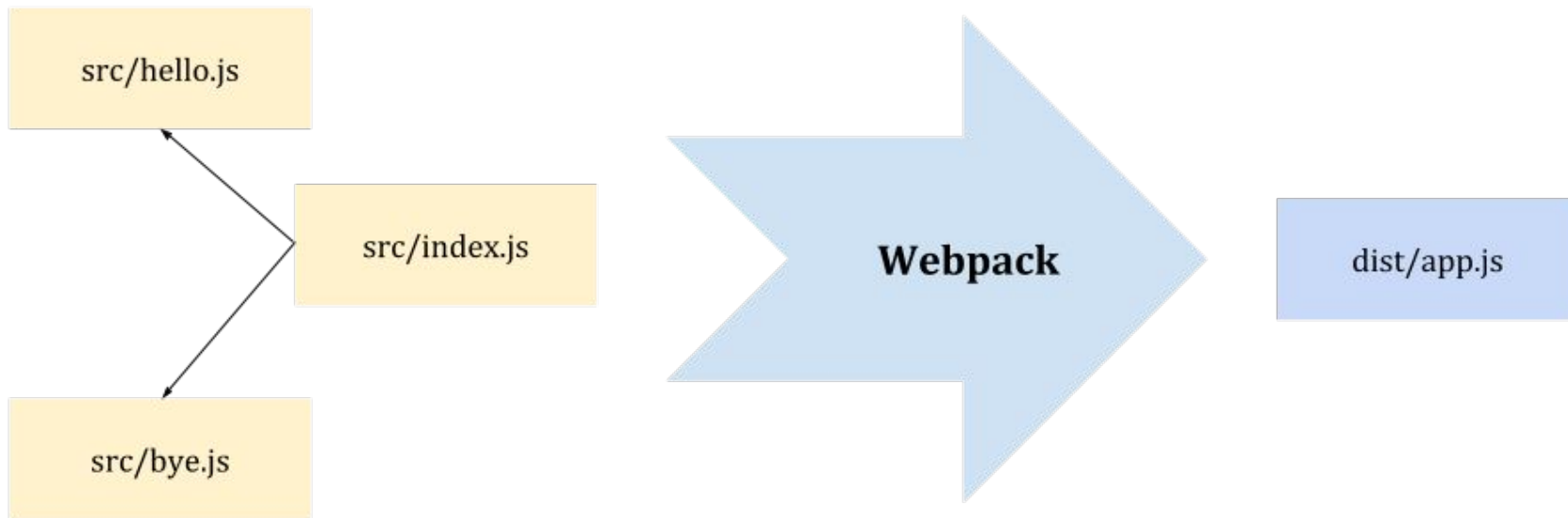
<script async type="text/javascript" src="http://www.googletagservices.com/tag/js/gpt.js"></script>
<script type="text/javascript" async src="http://stats.g.doubleclick.net/dc.js"></script>
<script type="text/javascript">...</script>
<script>...</script>
<script type="text/javascript">...</script>
<script type="text/javascript">...</script>
<script>...</script>
<script async type="text/javascript" src="http://partner.googleadservices.com/gpt/pubads_im
<link rel="stylesheet" type="text/css" href="/stdtheme.css">
<link type="text/css" rel="stylesheet" charset="UTF-8" href="https://translate.googleapis.c
<script type="text/javascript" charset="UTF-8" src="https://translate.googleapis.com/transl
<script src="http://www.google.com/jsapi?autoload=%7B%22modules%22%3A%5B%7B%22name%2...
ages%22%3A%5B%22search%22%5D%2C%22callback%22%3A%22_gcse.sacb%22%7D%5D%7D" type="text/java
<script type="text/javascript" charset="UTF-8" src="https://translate.googleapis.com/transl
<link type="text/css" href="http://www.google.com/uds/api/search/1.0/bdb5c6edfd66b31992bfba
<link type="text/css" href="http://www.google.com/cse/style/look/v2/default.css" rel="style
<script type="text/javascript" src="http://www.google.com/uds/api/search/1.0/bdb5c6edfd66b3
<script type="text/javascript" src="http://www.google.com/uds/api/ads/3.0/ef27064f0831a6e40
<style type="text/css">...</style>
<style type="text/css">...</style>
<script type="text/javascript" src="http://pagead2.googlesyndication.com/pagead/osd.js"></s
</head>

```

Vs

```
<script type="text/javascript" src="./app.js"/>
```

How does it work?



Demo 1

Most simple project structure, no configuration

Default entry and target



Configuration

- entry: src/index.js (default)
 - Entry is the source file where webpack will start processing.
 - Then it determines all the dependent files.
- output: dist/main.js (default)
 - All the dependent files are loaded by the loaders and then processed by plugins
 - After being processed they are written in the output file
- You can have multiple entry-output pairs.
- You can have multiple files in the same entry.
- Using variable, you can use the original file name: `[name].bundle.js`

Demo 2

Most basic configuration

Shows one entry and target

Terminologies

—



Loaders

- Loads your resources into webpack.
- i.e.: css-loader, babel-loader
- Called from rules-
 - Each rule has a test/condition
 - If a resource meets the condition, loader(s) is/are applied to the resource
 - If multiple loaders applied they are called loader chaining.
 - Loaders are applied from right to left or bottom to top (depending how often you press `enter`)
 - Webpack loads the resource in memory.

Demo 3

Configuration with loaders



Plugin

- Does everything that loaders can't do...
- Gets applied once the files are loaded.
 - There is no condition/test- gets applied on the entire package.*
 - * unless the plugin itself accepts condition as an argument/option.
 - You should pass an instance of plugin (i.e. `new MyAwesomePlugin(options)`)
- You can skip optimization/minification plugins for development to improve webpack build performance



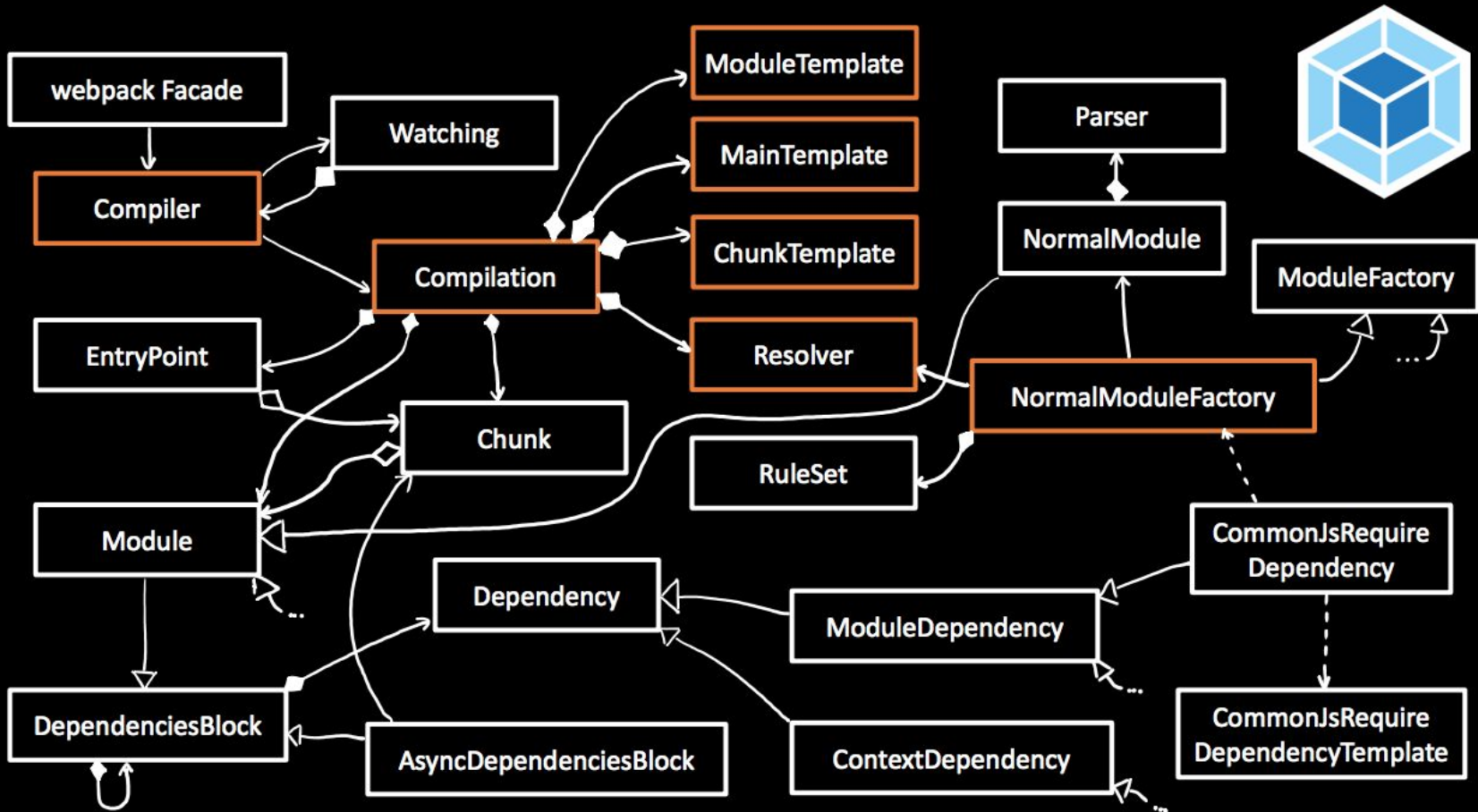
Other Important Config

- devtool: lets you choose how source maps are generated
- mode: tells webpack which optimizations to use
 - Options: none, development, production (default)
- externals: used for third party libraries (i.e. jQuery)
- optimization: built in optimization configurations
 - Lot of them used to be their own plugins, now they are part of the core
-

Demo 4

Configuration with plugins and few other options

How does webpack **REALLLLLLLY**
work?



Src: How works. Tobias Koppers
(link: <https://raw.githubusercontent.com/sokra/slides/master/data/how-webpack-works.pdf>)

WebPack 4 (legato)



What's New

- Zero Config
- `mode` property
- Some common plugins are removed and added to the core as configuration
- WebAssembly support

For full list: <https://github.com/webpack/webpack/releases/tag/v4.0.0>



Migration Tips

- Some configurations (loaders/plugins) have changed.
- Remove the unsupported plugins & re-configure optimizations
- Update the loaders configuration
- Make sure all loaders & plugins are taken care of
- Build!
- Compare with the previous builds: size, time (both production and development)
- Test your product!

Migration guide:

[https://github.com/webpack/webpack.js.org/blob/content/migration-4/src/content/guides/migrating.m](https://github.com/webpack/webpack.js.org/blob/content/migration-4/src/content/guides/migrating.md)
d



Resources

- Documentation: <https://webpack.js.org/concepts/>
- A really good article for beginners:
<https://medium.com/javascript-training/beginner-s-guide-to-webpack-b1f1a3638460>
- To understand nuts and bolts:
<https://medium.com/@rajaraodv/webpack-the-confusing-parts-58712f8fcad9>

Github: <https://github.com/mtanjir/webpack4demo>

Q&A
