∞ INFINUM

WORDPRESS – WEBPACK CAN THEY WORK (BUNDLE) TOGETHER?

IVAN RUŽEVIĆ
IVAN.RUZEVIC@INFINUM.CO

What will you get?

Why Webpack and WordPress?

WordPress Config overview.

Wp Boilerplate

Maybe it will help you

What should every WordPress project have?

CODING
STANDARDS

JAVASCRIPT

TASK
RUNNER

FONTS

IMAGES

VERSION
CONTROL

LINTERS

SASS

PRE-PROCESSOR

WHY WEBPACK?

Personal **Preferences!**

Currently IN

Speed

Code splitting

JavaScript

A Lot of Plugins

Easy to Extend

Module bundler

In WordPress

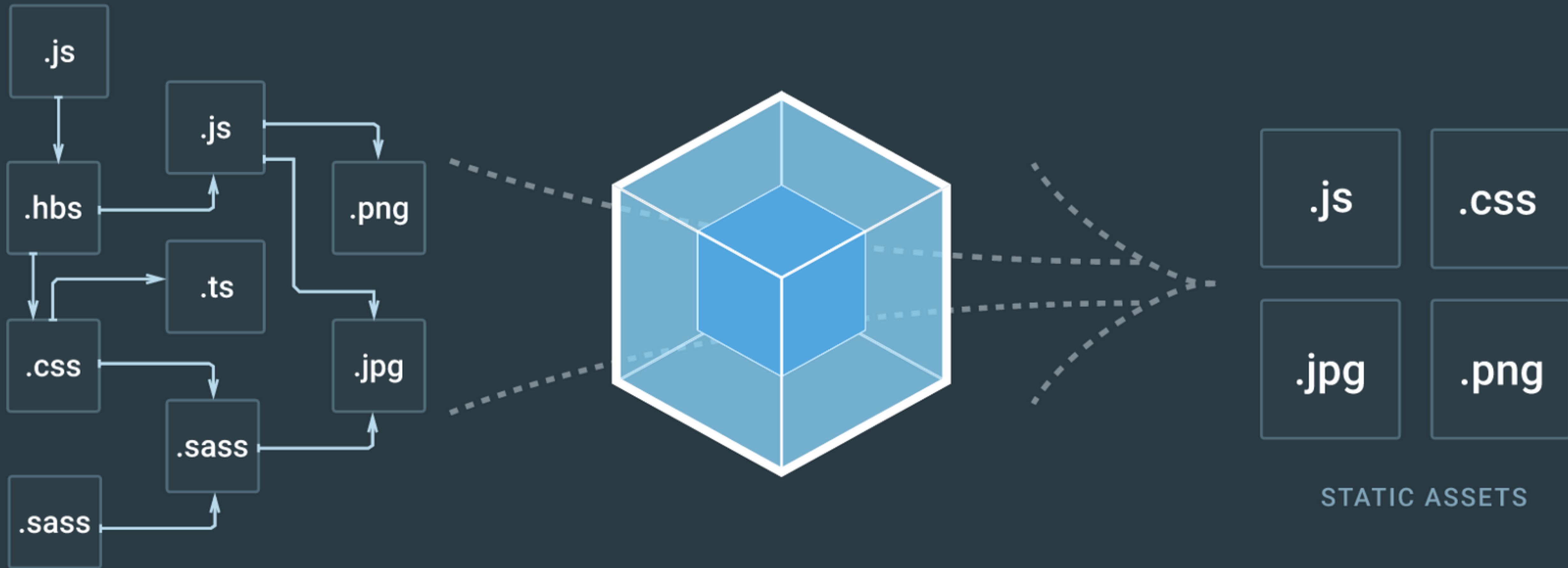All in one file (http2?)

Fast Build

Faster Code Execution (GPS)

Multiple options and features right out of the box

ES6

Wp Core

all sorts of awesome goodness

Webpack **build** process

.js

.js

.hbs

.png

.ts

.css

.jpg

.sass

.sass

.js

.css

.jpg

.png

STATIC ASSETS

MODULES WITH DEPENDENCIES

Love Hate

Webpack vs Gulp vs Grunt Comparison

	Webpack	Gulp	Grunt
Definition	Module bundler	Task runner	Task runner
Written in	Node.js	Node.js	Node.js
Monthly downloads	10.963,214	3.321,309	1.923,378
Used by	9GAG, Pinterest, Soundcloud, Flipboard, Airbnb, Udemy, 99	Netflix, FashionUnited, Myntra, MakerBot, AdRoll, Ekimetrics, Trigger,	Twitter, Adobe, jQuery, Mozilla, Bootstrap, Cloudant, Bitovi, Opera,

WebPack 4

⊙ config

Presets

WEBPACK CONFIG?

Require

```
const CleanWebpackPlugin = require('clean-webpack-plugin');
const ExtractTextPlugin  = require('extract-text-webpack-plugin');
const BrowserSyncPlugin  = require('browser-sync-webpack-plugin');
const CopyWebpackPlugin  = require('copy-webpack-plugin');
```

```javascript
const allModules = {
  rules: [
    {
      test: /\.(js|jsx)$/,
      use: 'babel-loader',
      exclude: /node_modules/,
    },
    {
      test: /\.json$/,
      exclude: /node_modules/,
      use: 'file-loader',
    },
    {
      test: /\.(png|svg|jpg|jpeg|gif|ico)$/,
      exclude: [/fonts/, /node_modules/],
      use: `file-loader?name=${outputImages}`,
    },
    {
      test: /\.(eot|otf|ttf|woff|woff2|svg)$/,
      exclude: [/images/, /node_modules/],
      use: `file-loader?name=${outputFonts}`,
    },
    {
      test: /\.scss$/,
      exclude: /node_modules/,
      use: ExtractTextPlugin.extract({
        fallback: 'style-loader',
        use: ['css-loader', 'postcss-loader', 'sass-loader'],
      }),
    },
  ],
};
```

Loaders For All

```javascript
const proxyUrl = 'dev.boilerplate.com';

const allPlugins = [
  new ExtractTextPlugin(outputCss),

  new webpack.ProvidePlugin({
    $: 'jquery',
    jQuery: 'jquery',
  }),

  // Use BrowserSync For assets
  new BrowserSyncPlugin({
    host: 'localhost',
    port: 3000,
    proxy: proxyUrl,
  })

  new webpack.DefinePlugin({
    'process.env': {
      NODE_ENV: JSON.stringify(process.env.NODE_ENV || 'development'),
    },
  }),

  // Is using vendor files, but prefered to use npm
  new CopyWebpackPlugin([{
    from: `${pathTheme}/assets/scripts/vendors`,
    to: `${pathTheme}/public/scripts/vendors`
  }])
];
```

Plugins For Everything

```
// Use only for production build
if (!DEV) {
  allPlugins.push(

    new CleanWebpackPlugin([themeOutput]),

    new webpack.optimize.UglifyJsPlugin({
      output: {
        comments: false,
      },
      compress: {
        warnings: false,
        drop_console: true,
      },
      sourceMap: true,
    })
  );
}
```

Plugins

For Production

```javascript
const outputJs = 'scripts/[name].js';
const outputCss = 'styles/[name].css';
const outputFile = '[name].[ext]';
const outputImages = `images/${outputFile}`;
const outputFonts = `fonts/${outputFile}`;

module.exports = [

  // Theme Skin
  {
    context: path.join(__dirname),
    entry: {
      application: [themeEntry],
      applicationAdmin: [themeAdminEntry],
    },
    output: {
      path: themeOutput,
      publicPath: themePublicPath,
      filename: outputJs,
    },

    module: allModules,

    plugins: allPlugins,

    devtool: DEV ? '#inline-source-map' : '',
  },
];
```

Export

Assets

WP BOILERPLATE

What is in it?

Watch and Build Scripts

Webpack 3

Linting

PHP Error Check

WPCS Style Guide

BrowserSync

PostCSS

Autoloader

Build and Pre-build Bash Script

GitIgnore

Sass Files Structure

Pre-created Theme Files

Functions structure

BEM Menus

Google rich snippets

Namespacing

Object-oriented

Open Source

BrowserSync

Webpack Dev Server

Cross Devices

Watching Assets

Watching PHP

Super Fast

```php
function get_assets_version( $filename = null ) {
 if ( ! $filename ) {
  return false;
 }


 $file = get_template_directory() . $filename;


 if ( ! file_exists( $file ) ) {
  return;
 }


 return filemtime( $file );
}
```

Cache busting

```php
$theme_name = 'test-scripts';
$main = '/skin/public/scripts/application.js';

wp_register_script(
 $theme_name,
 get_template_directory_uri() . $main,
 array(),
 get_assets_version( $main )
);

wp_enqueue_script( $theme_name );
```

CONCLUSION

04

WordPress + Webpack

Great Fit

A Lot of Benefits

Start building WordPress with this awesome tool!

Wp Boilerplate

https://github.com/infinum/wp-boilerplate

Headless WordPress

HELPFUL RESOURCES

webpack.js.org

webpack-when-to-use-and-why

gulp-vs-grunt-vs-webpack

postcss

browser-sync-webpack-plugin

wordpress-bem-menu

infinum-coding-standards-wp

∞ INFINUM

Any questions?

IVAN.RUZEVIC@INFINUM.CO
@ MUSTRA1

Visit infinum.co or find us on social networks:

 infinum.co   infinumco   infinumco   infinum