WEBPACKER

frontend development in rails 5.1

Paul Mölders
Frontend Developer - Freelancer
p0wl.space

ASSET PIPELINE FOR FRONTEND-HEAVY APPS

global namespace & gems for js packages

WEBPACKER

WEBPACK FOR RAILS

YARN FOR DEPENDENCIES

REACT / VUE / ANGULAR SETUP INCLUDED

INSTALL WEBPACKER

```
# Gemfile
gem 'webpacker', '~> 1.0' # or directly from github: 'rails/webpacker'

# New project
rails new myapp --webpack=react # react/angular/vue

# Existing project
rails webpacker:install
rails webpacker:install:react # react/angular/vue
```

webpacker binstubs

```
bin/webpack-dev-server  # http server to serve assets
bin/webpack-watcher     # file watcher
bin/webpack             # hooked into assets:precompile
```

webpacker related files

```
app/javascript # this is your frontend application root folder
app/javascript/packs/application.js # this is an entry, include this in your html and kickoff your app here
app/javascript/packs/hello_react.jsx # another entry, with react usage
```

application entry point
pack

```
// app/javascript/packs/hello_react.jsx
import React from 'react'
import ReactDOM from 'react-dom'

const Hello = props => (
  <div>Hello {props.name}!</div>
)

Hello.defaultProps = {
  name: 'David'
}

Hello.propTypes = {
  name: React.PropTypes.string
}

document.addEventListener('DOMContentLoaded', () => {
  ReactDOM.render(
    <Hello name="React" />,
    document.body.appendChild(document.createElement('div')),
  )
})
```

delivering webpacker output

```
# application.html.erb
<%= javascript_pack_tag 'hello_react' %>
<%= stylesheet_pack_tag 'hello_react' %>

# will become
<script src="/packs/hello_react-349bcb5c3feb6328691b.js"></script>
<link rel="stylesheet" media="screen" href="/packs/hello_react-dc02976b5f94b507e3b6.css">
```

webpacker configuration files

```
config/webpack/configuration.js
config/webpack/development.js
config/webpack/development.server.js # webpack-dev-server configuration
config/webpack/development.server.yml
config/webpack/loaders/assets.js     # yep
config/webpack/loaders/babel.js      # a
config/webpack/loaders/coffee.js     # file
config/webpack/loaders/erb.js        # for
config/webpack/loaders/react.js      # every
config/webpack/loaders/sass.js       # filetype
config/webpack/paths.yml
config/webpack/production.js         # production settings obviously
config/webpack/shared.js             # webpack config for all environments
```

CODE

WEBPACKER CAN COEXIST WITH THE ASSET PIPELINE

BUZZWORD CHECK

TREE SHAKING

> **Tree shaking is a term commonly used in the JavaScript context for dead-code elimination, or more precisely, live-code import.**

- *webpack.js.org/guides/tree-shaking*

tree shaking with webpacker

enabled by default (if you use react)

HMR

> **Hot Module Replacement (HMR) exchanges, adds, or removes modules while an application is running without a page reload.**

- webpack.js.org/concepts/hot-module-replacement

hmr with webpacker

not enabled, can be configured

CSS MODULES

> **A CSS Module is a CSS file in which all class names and animation names are scoped locally by default.**

- github.com/css-modules/css-modules

css modules with webpacker

not enabled, can be configured

RECAP

easy to start with

sensible defaults

full power over configuration

production ready

CRITIQUE

complexity increased with v1.1

breaking changes, upgrade path was ugly

>10 configuration files

thanks!

Paul Mölders

p0wl.space