



链滴

前端模块系统的发展史 -- Webpack

作者: [owsieman](#)

原文链接: <https://ld246.com/article/1532774921667>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

Webpack是什么

我们在使用React、Angular、Vue或者自定义的前端框架时，基本上都会使用到Webpack，因为你的模块形式是CommonJS、AMD还是一般的JS文件，或者你需要动态表达式`require("./"+name+".ade")`Webpack基本上都可以帮你解决。加上他的Loader转换器可以将其它各种非JavaScript模块都转换成JavaScript，并帮你处理。它不光有自己的API方便使用，还有一套完整的插件体系，开发可以通过插件进行快速的开发，并且大部分插件都是开源的。

通过配置文件来了解Webpack

虽然从webpack v4.0.0开始，可以不用引入一个配置文件，但是要真正了解webpack还是需要从它配置文件说起。

这里给一个例子，有些注释写的不对的地方请大家指出：

```
const path = require('path');

module.exports = {
  mode: "production", // 这个是最优化的编译结果
  mode: "development", // 开发模式，可以使用很多的工具来增强开发
  mode: "none", // 什么也不做，注意它不是默认值

  //入口文件的位置
  entry: "./app/entry",
  entry: ["./app/entry1", "./app/entry2"],
  entry: {
    a: "./app/entry-a",
    b: ["./app/entry-b1", "./app/entry-b2"]
  },

  //输出路径
  output: {

    path: path.resolve(__dirname, "dist"), // string

    filename: "bundle.js", //单页的输出
    filename: "[name].js", // 多页的输出
    filename: "[chunkhash].js", //文件缓存（长期的），原来的文件名上带一串hash字符串，自己测只适用于js和css
    publicPath: "/assets/", //资源路径
    publicPath: "",
    publicPath: "https://cdn.example.com/",

    library: "MyLibrary",

    //这里可以参见前面一篇文章
    libraryTarget: "umd",
    libraryTarget: "umd2",
    libraryTarget: "commonjs2",
    libraryTarget: "commonjs",
    libraryTarget: "amd",
    libraryTarget: "this",
    libraryTarget: "var",
```

```

libraryTarget: "assign",
libraryTarget: "window",
libraryTarget: "global",
libraryTarget: "jsonp",

chunkFilename: "[id].js",
chunkFilename: "[chunkhash].js",

jsonpFunction: "myWebpackJsonp",

//源文件的映射, 方便调试
sourceMapFilename: "[file].map",
sourceMapFilename: "sourcemap/[file].map",

devtoolModuleFilenameTemplate: "webpack:///[resource-path]",

devtoolFallbackModuleFilenameTemplate: "webpack:///[resource-path]?[hash]",

//如果output.libraryTarget设置为umd, `output.library`设置了, 这个选项设为true将会给命名
amd模块`
umdNamedDefine: true, // boolean
// use a named AMD module in UMD library

//选项用于跨域加载chunks
crossOriginLoading: "use-credentials", //这里是一个枚举选项, 允许跨域加载, 会在请求中发
证书
crossOriginLoading: "anonymous", //匿名, 不发送证书
crossOriginLoading: false, //设为false禁止跨域加载

//模块的配置项
module: {
  rules: [
    // 这里使用我换成常用的一些loader
    {
      test: /\.js$/,

      enforce: 'pre',

      loader: 'eslint-loader',

      options: {
        emitWarning: true,

        fix: true

      },
    },
  ],
},

//使用scss-loader | css-loader和style-loader

```

```
{
  test: /\.css$/,
  // exclude: /node_modules/,
  use: plugin.extract({
    use: [{
      loader: 'css-loader',
      options: {
        module: true,
        imporLoader: 1,
        sourceMap: true,
      },
    },
  ],
  fallback: 'style-loader',
}),
},

//对图片和字体进行打包
{
  test: /\.(gif|png|jpg|woff|svg|eot|ttf)\{?.*$/,
  loader: 'url-loader?limit=100&name=resource/[name].[ext]'
},
```

```

    ],
  },
  //这个模块解决模块的请求, 他不适用于loader
  resolve: {

    modules: [
      "node_modules",
      path.resolve(__dirname, "app")
    ],

    extensions: [".js", ".json", ".jsx", ".css"],
    //这里用来设置别名
    alias: {

      "module": "new-module",

      "only-module$": "new-module",

      "module": path.resolve(__dirname, "app/third/module.js"),
    },
  },
  alias: [
    {
      name: "module",
      // the old request
      alias: "new-module",
      // the new request
      onlyModule: true
      // if true only "module" is aliased
      // if false "module/inner/path" is also aliased
    }
  ],
  symlinks: true,
  // follow symlinks to new location
  descriptionFiles: ["package.json"],
  // files that are read for package description
  mainFields: ["main"],
  // properties that are read from description file
  // when a folder is requested
  aliasFields: ["browser"],
  // properites that are read from description file
  // to alias requests in this package
  enforceExtension: false,
  // if true request must not include an extensions
  // if false request may already include an extension
  moduleExtensions: ["-module"],
  enforceModuleExtension: false,
  // like extensions/enforceExtension but for module names instead of files
  unsafeCache: true,
  unsafeCache: {},

  cachePredicate: (path, request) => true,

```

//插件的使用,比如压缩, 处理网页模板等插件的使用, 大家可以去更多的到网上去查查, 这里就不赘述

```
plugins: [  
  // ...  
]  
  
}  
};
```

这里基本上算是将前端模块系统的发展史给介绍完了, 如果有写的不是很清楚或者不正确的地方, 希望大家能够提出。

接下来, 我将写一系列的Angular4+的文章, 大家可以继续关注。

最后, 我再给大家一个CSS的蜡烛动画的代码, 这里不给出效果, 大家自己动手去试一下, 算是对之前css动画系列的一个补充:

```
<div class="candle">  
  <span class="glow"></span>  
  <span class="flames"></span>  
  <span class="thread"></span>  
</div>
```

```
<style>  
body{  
  background: #000510;  
  font-family: 'Titillium Web', sans-serif;  
  width: 100%;  
  width: 150em;  
  height: 40em;  
}  
.candle {  
  width: 15em;  
  height: 30em;  
  font-size: 7px;  
  background: linear-gradient(  
    orange,  
    darkorange,  
    sienna,  
    saddlebrown 50%,  
    rgba(0, 0, 0, 0.6)  
  );  
  box-shadow:  
    inset 2em -3em 5em rgba(0, 0, 0, 0.4),  
    inset -2em 0 5em rgba(0, 0, 0, 0.4);  
  border-radius: 10em / 4em;  
  position: relative;  
  display: flex;  
  justify-content: center;  
  top: 10em;  
}
```

```
.candle::before {
  content: "";
  position: absolute;
  width: inherit;
  height: 5em;
  border: 0.2em solid darkorange;
  border-radius: 50%;
  box-sizing: border-box;
  background: radial-gradient(
    #444,
    orange,
    saddlebrown,
    sienna,
    darkorange
  );
  filter: opacity(0.7);
}

.thread {
  position: absolute;
  width: 0.6em;
  height: 3.6em;
  top: -1.8em;
  background: linear-gradient(
    #111,
    black,
    orange 90%
  );
  border-radius: 40% 40% 0 0;
}

.flames {
  position: absolute;
  width: 2.4em;
}

.flames::before {
  content: "";
  position: absolute;
  width: inherit;
  height: 6em;
  background-color: royalblue;
  top: -4.8em;
  border-radius: 50% 50% 35% 35%;
  border: 0.2em solid dodgerblue;
  box-sizing: border-box;
  filter: opacity(0.7);
}

.flames::after {
  content: "";
  position: absolute;
  width: inherit;
}
```

```
height: 12em;
top: -12em;
background: linear-gradient(white 80%, transparent);
border-radius: 50% 50% 20% 20%;
box-shadow: 0 -0.6em 0.4em darkorange;
animation:
  enlarge 5s linear infinite,
  move 6s linear infinite;
}
```

```
@keyframes move {
  0%, 100% {
    transform: rotate(2deg);
  }

  50% {
    transform: rotate(-2deg);
  }
}
```

```
@keyframes enlarge {
  0%, 100% {
    height: 12em;
    top: -12em;
  }

  50% {
    height: 14em;
    top: -13em;
  }
}
```

```
.glow {
  position: absolute;
  width: 10em;
  height: 18em;
  background-color: orangered;
  border-radius: 50%;
  top: -17em;
  filter: blur(6em);
  animation: blink 100ms infinite;
}
```

```
@keyframes blink {
  to {
    filter: blur(6em) opacity(0.8);
  }
}
</style>
```