


Webpack | vite | Rollup | Gulp | Grunt 比较

原创 前端阿彬 已于 2023-04-03 10:58:54 修改 阅读量2.7k 收藏 13 点赞数 1

分类专栏: 前端实用知识学习 文章标签: webpack gulp 前端 vite 打包工具

 前端实用知识学习 专栏收录该内容

17 订阅 72 篇文章

前言

模块化管理工具和自动化构建工具是不同的。两者主要体现在侧重点不一样。自动化构建工具侧重于 **前端开发** 的 **整个过程** 的控制管理（像是 **流水线**）而模块化管理工具更侧重于**模块打包**，当然我们可以把开发中的所有资源（图片、js文件、css文件等）都可以看成模块。

• Webpack

webpack可以说是当前最流行的**模块化管理工具**和打包工具。其通过loader的转换，可以将任何形式的资源视作模块。它还可以将各个模块通过系打包成符合生产环境部署的前端资源。它还可以将应用程序分解成可管理的代码块，可以按需加载。

打包原理：

解析各个模块的依赖关系 使用loader转换文件，使用plugin注入钩子，打包合并模块，最终生成bundle文件，使用express开启本地服务器， 浏览器调bundle文件，也就是打包过后的文件。

优点：

- 1. 基本之前gulp 可以进行的操作处理，现在webpack也都可以做；
- 2. 同时支持热更新，支持tree shaking 、Scope Hoisting、动态加载、代码拆分、文件指纹、代码压缩、静态资源处理等；
- 3. 支持多种打包方式。

等等

缺点：

- 1.各个模块之间的依赖关系过于复杂 会导致打包速度很慢
- 2.使用 **热更新** 时，改动一个模块，其他有依赖关系的模块也会重新进行打包
- 3.不支持打包出esm格式的代码 (打包后的代码再次被引用时tree shaking 困难)， 打包后冗余代码较多。

• Vite

和webpack差不离，vite是当下新兴的**模块化管理工具**和打包工具。它的本地启动速度相较于webpack快了很多，但是目前来说，vite还完全没有换webpack的能力，不管是从社区还是从能力来讲，vite他本身还是太过脆弱，他的产生跟火热完全依赖于vue本身的热度。

打包原理：

使用koa开启本地服务器，没有webpack那样打包合并的过程，所以启动服务器很快，@vue/compiler-sfc会对模块进行编译（修改一些引入文件的路成js字符串）浏览器再请求编译好的模块。

缺点：

- 1.项目的开发浏览器要支持esmodule
- 2.不能识别commonjs语法
- 3.生态不及webpack，加载器、插件不够丰富
- 4.生产环境esbuild构建对于css和代码分割不够友好
- 5.没被大规模重度使用，会隐藏一些问题

觉得还不错？



前端阿彬

关注

1



• Rollup

Rollup 是一个**模块打包工具**，可以将我们按照 ESM (ES2015 Module) 规范编写的源码构建输出如下格式：

- IIFE: 自执行函数, 可通过 `<script>` 标签加载
- AMD: 通过 RequireJS 加载
- CommonJS: Node 默认模块规范, 可通过 Webpack 加载
- UMD: 兼容 IIFE, AMD, CJS 三种模块规范
- ESM: ES2015 Module 规范, 可用 Webpack, Rollup 加载

优点：

1. 支持动态导入。
2. 支持tree shaking。仅加载模块里用得着的函数以减小文件大小。
3. Scope Hoisting。rollup可以将所有小文件生成到一个大文件中，所有代码都在同一个函数作用域里，不会像 Webpack 那样用很多函数来包装
4. 没有其他冗余代码，执行很快。除了必要的 `cjs`, `umd` 头外，bundle 代码基本和源码差不多，也没有奇怪的 `__webpack_require__`, `Object.defineProperty` 之类的东西，

缺点：

1. 不支持热更新功能；
2. 对于commonjs模块，需要额外的插件将其转化为es2015供rollup 处理；
3. 无法进行公共代码拆分。

适用场景：

由纯js开发的第三方库；需要生成单一的umd文件的场景

比较（和webpack）：

- 基于权衡，Rollup 目前还不支持代码拆分（Code Splitting）和模块的热更新（HMR）
- 一般而言，**对于应用使用 Webpack**，**对于类库使用 Rollup**；
- 需要代码拆分(Code Splitting)，或者很多静态资源需要处理，又或者构建的项目需要引入很多 CommonJS 模块的依赖时，使用 webpack。代码ES6 模块，而且希望代码能够被其他人直接使用，使用 Rollup
- Rollup 与 Webpack 有着不同的用途，因此会共同存在，并相互支持
- React 已经将构建工具从 Webpack 换成了 Rollup

使用：

<https://www.cnblogs.com/tughenhu0707/p/8179686.html>

• Gulp

Gulp 是基于“流”的**前端自动化构建工具**，采用代码优于配置的策略，更易于学习和使用，它让简单的任务简单，复杂的任务可管理。

它是基于Nodejs，自动化地完成 javascript、coffee、sass、less、html/image、css 等文件的测试、检查、合并、压缩、格式化、浏览器自动刷文件生成，并监听文件在改动后重复指定的这些步骤。

优点：

1. gulp文档简单，学习成本低，使用简单；
2. 对大量源文件可以进行流式处理，借助插件，可以对文件类型进行多种操作处理。

缺点：

1. 不支持tree-shaking、热更新、代码分割等；
2. gulp 对 js 模块化方案无能为力，只是对静态资源

觉得还不错？



前端阿彬

关注

1 1 1

适用场景：

静态资源密集操作型场景，主要用于css、图片等静态资源的处理操作。

比较（和grunt）：

- 1. 易用 。Gulp相比Grunt更简洁，而且遵循代码优于配置策略，维护Gulp更像是写代码。
- 2. 高效 。Gulp相比Grunt更有设计感，核心设计基于Unix流的概念，通过管道连接，不需要写中间文件。
- 3. 高质量。Gulp的每个插件只完成一个功能，这也是Unix的设计原则之一，各个功能通过流进行整合并完成复杂的任务。例如：Grunt的imagemin压缩图片，同时还包括缓存功能。他表示，在Gulp中，缓存是另一个插件，可以被别的插件使用，这样就促进了插件的可重用性。目前官方列出插件。
- 4. 易学。Gulp的核心API只有5个，掌握了5个API就学会了Gulp，之后便可以通过管道流组合自己想要的任务。
- 5. 流 。使用Grunt的I/O过程中会产生一些中间态的临时文件，一些任务生成临时文件，其它任务可能会基于临时文件再做处理并生成最终的构建后使用Gulp的优势就是利用流的方式进行文件的处理，通过管道将多个任务和操作连接起来，因此只有一次I/O的过程，流程更清晰，更纯粹。
- 6. 代码优于配置 。维护Gulp更像是写代码，而且Gulp遵循CommonJS规范，因此跟写Node程序没有差别。

使用：

快速入门 · gulp.js 中文文档

• Grunt

Grunt 是一套前端自动化工具，帮助处理反复重复的任务。一般用于：编译，压缩，合并文件，简单语法检查等

特点：

- Grunt有一个完善的社区，插件丰富，根据社区的结果显示，共计3,439个插件，其中49个官方插件。
- 它简单易学，你可以随便安装插件并配置它们。许多常见的任务都有现成的Grunt插件，而且有众多第三方插件，如：CoffeeScript, Handle Jade, JsHint, Less, RequireJS, Sass, Styles。而且通过参考文档进行配置便可以使用。
- 你不需要多先进的理念，也不需要任何经验。

使用：

快速入门 - Grunt: JavaScript 世界的构建工具 | Grunt 中文网

grunt-webpack：将webpack集成到grunt构建过程中

咕WebWebpack 将Webpack与Grunt一起使用。要求 grunt-webpack版本4支持webpack 4以及版本5 beta。如果需要，请使用webpack-dev-server 3.2.0版本（可选）。

认识webpack、理解webpack与grunt、gulp的核心区别01

钢铁

认识webpack 什么是webpack 官方解释 At its core,webpack is a static module bundler for modern JavaScript application. 即 webpack是一个现代的JavaScript应用的静态

尚硅谷webpack课程学习笔记_cannot find module 'ajv

1、安装好之后建一个webpack的配置文件,在根路径下,并且文件名必须是webpack.config.js 2、添加基本配置 3、我们之前打包执行的是npm webpack ./src/main.js --mode

webpack基础_webpack基础csdn

Webpack 是一个静态资源打包工具,目前市面上最流量的是 Webpack,其他的打包工具还有Grunt、Glup、Parcel、Rollup、Vite...等。 Webpack 有什么用? 总的来说:将工

前端构建工具对比 webpack、vite、esbuild等 最新发布

u012764358的

人类和动物最大的区别是会使用工具，工具的作用就是提升我们的生产效率，同样，随着前端工程化的演进，伴随着一些优秀的工程化的生产效率工具，今天就来聊一聊，

project-starter:Webpack | 打字稿| SASS | 角度的

项目启动器 用法 \$ npm start运行webpack-dev-server \$ npm test运行单元测试 \$ npm run lint或gulp lint到lint代码 \$ npm run build或gulp来构建 \$ gulp release:[major]minc

前端构建简史_前端构建工具进化历程

webpack webpack相对于传统的glup等传统构建工具对构建流程和资源有更高层次的抽象,可以将所有类型资源的构建过程进行统一的管理,整体的分析和优化。

构建工具:grunt、Glup、webpack

grunt、gulp、webpack grunt 1.官网:https://www.gruntjs.net/ 2.参考文档:http://www.cnblogs.com/wangfupeng1988/p/4561993.html 3.常用的插件: * grunt-contrib-clean—

webpack&vite 前端构建工具源码

webpack&vite 前端构建工具源码

觉得还不错? -

Webpack 和 Rollup：一样但又不同



前端阿彬

关注

1

