



构建性能： 减少编译范围

LazyCompilation 异步引用模块的按需编译

require('./xxx') 或 import('./xxx') 不会立即编译

lazyCompilation.js

```
module.exports = (base) => {  
  Object.assign(base, {  
    experiments: {  
      lazyCompilation: true,  
    },  
  });  
  return base;  
};
```

TypeScript

约束 Loader 执行范围 – exclude

Webpack 在处理 `node_modules` 中的 js 文件时会直接跳过这个 `rule` 项，不会为这些文件执行 Loader 逻辑。

config/exclude.js

```
module.exports = (base) => {  
  base.module.rules.map((v) => {  
    v.exclude = /node_modules/;  
  });  
  return base;  
};
```

JavaScript

约束 Loader 执行范围

对已经打包集成后的代码，无需在次编译

- noParse 跳过文件编译

提前打包的无需编译

- 前提条件是不包含 import 内容
- 无法 Treeshaking
- 必须保证程序无语法错误

vue 就不符合这个特点

Vue3 node_modules/vue/dist/vue.esm-bundle.js

webpack.config.js

```
const path = require("path");
module.exports = (base) => {
  base.module.noParse = /^(^vue$)|(^pinia$)|(^vue-router$)/;
  return base;
};
```

JavaScript