#### Angular脱坑记 番外编

##### 之记一次排错经历

问题描述

angular 6.1.5

rx-js 6.1.0

rxjs-compat 6.1.0

rx-js全称the React eXtension for JavaScript。一个重要的作用是提供异步调用接口，其定义了对象Observable、Subscription等。

故障现象：

测试环境编译、服务正常

生产环境编译通过、全称无报错

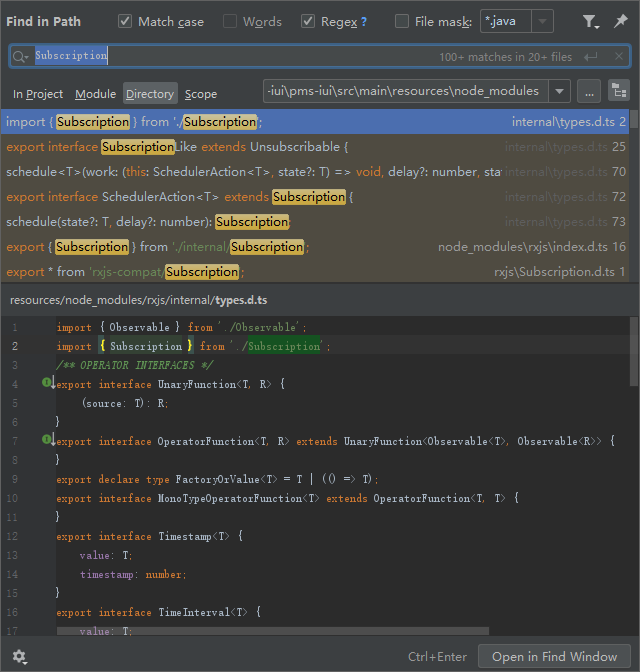
生产环境服务，访问页面时控制台报错“XXX: Subscription is not defined.”

解决

1. 验证故障现象
2. 检查页面

在故障页面右键选“检查”，观察到页面渲染至<app-root></app-root>，故推测为依赖项相关错误。

1. IDEA编辑器全局搜索Subscription对象，结果如下：



以上结果说明：

故障是依赖项rx-js导致的。

1. 检查package.json

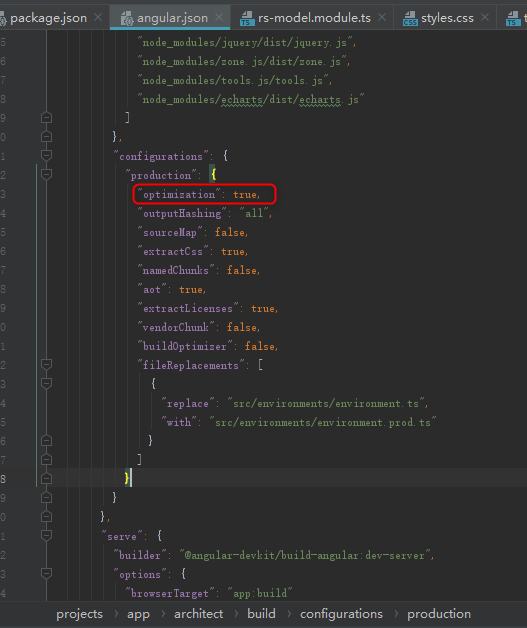
package.json

"rxjs": "~6.1.0",  
"rxjs-compat": "~6.1.0",

与npm版本比较，可知其版本号低于最新版本。将两个依赖项升级至6.3.0即可。

1. 关闭代码压缩选项

关闭optimization选项，在浏览器中可见原始JS代码（未混淆、压缩）



1. 进一步调试

进一步调试发现，angular-cli的@angular-devkit/build-optimizer与现有依赖库函数之间存在未知的副作用，故将其关闭：



1. 验证

部署生产环境Nginx服务，经测试，故障已解决。

附录build-optimizer扩展选项：

export interface BuildOptimizerOptions {

  content?: string;

  inputFilePath?: string;

  outputFilePath?: string;

  emitSourceMap?: boolean;

  strict?: boolean;

  isSideEffectFree?: boolean;

}

【小结】

1. build-optimizer可能存在副作用（Side Effect）
2. 关闭optimization功能有利于在生产环境调试依赖项
3. 如无特殊需要，尽量不要使用不成熟的新技术

□