

npx 是什么东西，跟 npm 又是什么关系呢？

最近一系列的文章不是 npm 就是 yarn，今天又来一个 npx。说到 npx 和 npm 啥关系呢？一个麦当劳，一个麦当娜？或是雷锋与雷锋塔的故事？其实都不是啊。前面两者真没有什么关系。但是 npx 和 npm 真就是兄嘢。

npx 介绍

npm 升级到 5.2.0 之后的人可能会注意到它时候发现会买一送一，自动安装了 **npx**。**npx**，类似于 npm 简化了项目开发中的依赖安装与管理，该工具致力于提升开发者使用包提供的命令行的体验。npx 允许我们使用本地安装的命令行工具而不需要再定义 npm run-script，并且允许我们仅执行一次脚本而不需要再将其实际安装到本地。

npx 的使用

在过去的几年中，npm 生态系统越来越倾向于将工具安装为项目本地 devDependencies，而不是要求用户在全局范围内安装它们。这意味着，像工具 mocha，grunt 和 bower，曾经被全球主要安装在系统上，现在可以对每个项目的基础上管理自己的版本。这也意味着，要使基于 npm 的项目启动并运行，您需要做的就是确保在系统上有 node + npm，克隆 git repo，然后 npm it 运行 install 和运行 test。由于 npm run-script 在路径中添加了本地二进制文件，因此工作正常！

缺点是，这使您无法以交互方式快速/方便地调用本地二进制文件。有几种方法

可以做到这一点，他们都有的一些烦恼：你可以添加这些工具 scripts，但是你需要记住通过使用传递参数--，你可以做类似的 shell 技巧 alias npx=PATH=\$(npm bin):\$PATH，或者你可以手动路径到他们./node_modules/.bin/mocha。这些都有效，但没有一个是理想的。npx 为我们提供了更好的方式。

总结一下全局安装 cmd 的弊端：

- 1、 对于不常用的 cmd，只是用一次。但一直占用我们的内存。
- 2、 现在的大多数项目都有自己的依赖管理。也就是每个项目使用的依赖或者指令都有自己的版本。一旦使用全局依赖进行调用会报错。

对于 npx 我们可以直接调用 npx command 通过这种方式直接调用指令。会非常方便。

举例来说：之前我们通过 npm run scripts 的方式运行 webpack 可以到 ./node_modules/.bin/webpack 这里找指令。对于 npx

只要本地已经安装过之后，直接执行 npx webpack 就可以。是不是非常方便。

对于没有的指令。包括 npx 的指令查找方式是怎么样的，接下来我们来看看。

当我们通过 npx 执行命令时：

npx 先从当前目录下 /node_modules/.bin 目录进行查找，如果查找到，直接把可执行文件进行执行。如果没有找到，会到全局的 Path 环境变量中配置的目录去查找全局指令。

如果在没找到 npx 可以自己下载该指令。并且执行。这样的话就避免了安装包一直占用全局内存的问题，而且每次下载都是最新的。

关于 npx 还有一些其他的内容，这里就不过多描述，想了解的同学可以去看看

官方文档。

最后和大家说一句：npx 大家可以理解成一个执行器。寻找路径分为三层：

当前路径下的 node_modules

环境变量下的 Path 变量的路径下的 node_modules（一般指全局）

如果没找到指令会进行安装。

