
更好用的依赖包管理工具 yarn

依赖包管理工具 npm 大家都知道，基本的常用指令大家也都了解，这里就不多说了。我们直接开门见山说一说，**npm 有哪些小问题**。其实用了这么久大家多少也有些感触。Npm 用起来很方便。但是 npm 的本源在国外，国内用户使用起来会非常不方便，不方便之处在于下载慢。然后我们会有 cnpm 去替代，但是 cnpm 会导致包丢失，下载的依赖包可能会出现问题，等等若干不确定情况。

下载速度慢是 npm 存在的问题之一。还有一个很大的问题是在于 npm 在同一项目中，对于依赖包的安装无法保持一致性。举个例子来说明：

在我们的项目中通常存在 package.json 文件用于记录相关依赖，但是里面记录的依赖的表现形式也不统一。

"5.0.3",

"~5.0.3",

"^5.0.3"

"5.0.3"表示安装指定的 5.0.3 版本，"~ 5.0.3"表示安装 5.0.X 中最新的版本，"^5.0.3"

表示安装 5.X.X 中最新的版本。这就麻烦了，常常会出现同一个项目，有的同事安装依赖后是 OK 的，有的同事会由于安装的版本不一致导致出现 bug。因为随着时间的流逝，依赖包存在着升级和变动。有些依赖包可能变化不大，基本引用的依赖包也不会发生变化，但是存在一些依赖包里面的相关依赖发生变化时，我们就无法正常使用了。

安装的时候，包会在同一时间下载和安装，中途某个时候，一个包抛出了一个错误，但是 npm 会继续下载和安装包。因为 npm 会把所有的日志输出到终端，有关错误包的错误信息就会在一大堆 npm 打印的警告中丢失掉，并且你甚至永远

不会注意到实际发生的错误。

根据上面我们吐槽的内容得知，影响我们心情的两大原因是：

npm 下载依赖包的网路问题

npm 版本不一致的痛点

说了有些痛点，那么存不存在解决这些痛点的方法呢，当然存在一方面 npm 自己也在不断更新，另一方面就是我们今天要说的 **yarn**。

什么是 Yarn

“Yarn 是由 Facebook、Google、Exponent 和 Tilde 联合推出了一个新的 JS 包管理工具，正如官方文档中写的，Yarn 是为了弥补 npm 的一些缺陷而出现的。他并没有完全取代 npm。这句话意味着，npm 还是有用的，不是说回了 Yarn 就不用 npm 了！。

Yarn 的优点

- **速度快**。速度快主要来自以下两个方面：
 1. **并行安装**：无论 npm 还是 Yarn 在执行包的安装时，都会执行一系列任务。npm 是按照队列执行每个 package，也就是说必须要等到当前 package 安装完成之后，才能继续后面的安装。而 Yarn 是同步执行所有任务，提高了性能。
 2. **离线模式**：如果之前已经安装过一个软件包，用 Yarn 再次安装时之间从缓存中获取，就不用像 npm 那样再从网络下载了。
- **安装版本统一**：为了防止拉取到不同的版本，Yarn 有一个锁定文件（lock file）记录了被确切安装上的模块的版本号。每次只要新增了一个模块，Yarn 就会创建（或更新）yarn.lock 这个文件。这么做就保证了，每一次拉取同一个项目依赖时，使用的都是一样的模块版本。npm 其实也有办法实现处处使用相同版本的 packages，npm5 以上会生成一

个锁定文件，在执行 `npm install` 的时候，该锁定文件会先被读取，和 Yarn 读取 `yarn.lock` 文件一个道理。

- **更简洁的输出：**npm 的输出信息比较冗长。在执行 `npm install` 的时候，命令行里会不断地打印出所有被安装上的依赖。相比之下，Yarn 简洁太多：默认情况下，结合了 emoji 直观且直接地打印出必要的信息，也提供了一些命令供开发者查询额外的安装信息。
- **多注册来源处理：**所有的依赖包，不管他被不同的库间接关联引用多少次，安装这个包时，只会从一个注册来源去装，要么是 npm 要么是 bower，防止出现混乱不一致。
- **更好的语义化：**yarn 改变了一些 npm 命令的名称，比如 `yarn add/remove`，感觉上比 npm 原本的 `install/uninstall` 要更清晰。

yarn 指令介绍

如何安装 yarn?

方法一： `npm install yarn -g`

方法二： 去官网下载 yarn 安装包（推荐）

官网地址： <https://yarnpkg.com/zh-Hans/docs/install#windows-stable>

一般来说，不推荐通过 `npm` 安装 `Yarn`，在用基于 `Node` 的包管理器安装 `Yarn` 时，该包未被签名，并且只通过基本的 `SHA1` 散列进行唯一完整性检查。这在安装系统级应用时有安全风险。因为这些原因，高度推荐用你的操作系统最适合的方式来安装 `Yarn`

Yarn 常用指令有哪些？与 npm 指令对比

Npm	Yarn	功能描述
<code>npm install</code>	<code>yarn install</code>	根据 <code>package.json</code> 安装所有依赖
<code>npm install --save</code>	<code>Yarn add</code>	添加依赖包
<code>npm install --save-dev</code>	<code>Yarn add -dev</code>	添加依赖包至 <code>devDependencies</code>
<code>npm install -g</code>	<code>Yarn global add</code>	全局安装依赖包

npm uninstall	Yarn remove	移除依赖包
npm run	yarn run	运行 package.json 中预定义的脚本

相比 Npm 的默认配置，Yarn 获赞颇多。用其可以方便生成锁文件，安装依赖非常迅速，且会自动添加进 package.json，同时安装与使用 Yarn 的成本也极小，使得 yarn 成为新起之秀。

