# NPM使用介绍

nodejs的出现，可以算是前端里程碑式的一个事件，它让前端攻城狮们摆脱了浏览器的束缚，踏上了一个更加宽广的舞台。前端的可能性，从此更加具有想象空间。

随着一系列基于nodes的应用/工具的出现，工作中与nodejs打交道的机会越来越多。无论在node应用的开发，还是使用中，包管理都扮演着一个很重要的作用。NPM（node package manager），作为node的包管理工具，极大地便利了我们的开发工作，很有必要了解一下。

## NPM是什么

NPM（node package manager），通常称为node包管理器。顾名思义，它的主要功能就是管理node包，包括：安装、卸载、更新、查看、搜索、发布等。

npm的背后，是基于couchdb的一个数据库，详细记录了每个包的信息，包括作者、版本、依赖、授权信息等。它的一个很重要的作用就是：将开发者从繁琐的包管理工作（版本、依赖等）中解放出来，更加专注于功能的开发。

NPM是随同NodeJS一起安装的包管理工具，能解决NodeJS代码部署上的很多问题，常见的使用场景有以下几种：

1.允许用户从NPM服务器下载别人编写的第三方包到本地使用。

2.允许用户从NPM服务器下载并安装别人编写的命令行程序到本地使用。

3.允许用户将自己编写的包或命令行程序上传到NPM服务器供别人使用。

由于新版的nodejs已经集成了npm，所以之前npm也一并安装好了。同样可以通过输入 "npm -v" 来测试是否成功安装。命令如下，出现版本提示表示安装成功:

npm -v

4.6.1

如果你安装的是旧版本的 npm，可以很容易得通过 npm 命令来升级，命令如下：

npm install npm -g

## 使用 npm 命令安装模块

nmp的包安装分为本地安装（local）、全局安装（global）两种，从命令来看，差别多了一个-g而已，比如：

npm install <Module Name> # 本地安装

npm install <Module Name> -g # 全局安装

二者之间有什么区别呢？

本地安装

1. 将安装包放在 ./node\_modules 下（运行 npm 命令时所在的目录），如果没有 node\_modules 目录，会在当前执行 npm 命令的目录下生成 node\_modules 目录。

2. 可以通过 require() 来引入本地安装的包。

全局安装

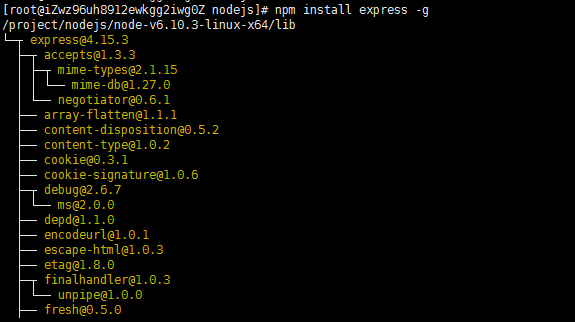
1. 将安装包放在 /usr/local 下或者你 node 的安装目录。

2. 可以直接在命令行里使用。

接下来使用全局方式安装express

npm install express -g

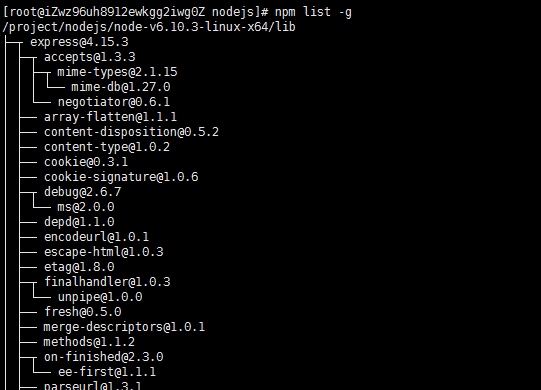
安装过程输出如下内容，第一行输出了模块的版本号及安装位置。



## 查看安装信息

你可以通过下面命令，查看所有全局安装的模块：

npm list -g 或者npm ls -g



如果要查看某个模块的版本号，可以使用命令如下：



总结一下：使用命令，如果不加-g，在当前目录下查找，如果加上-g，在全局安装的目录下查找。

## 使用package.json

package.json 位于模块的目录下，用于定义包的属性。此处不仔细讲，有兴趣，可自行查看。

## 卸载模块

我们可以使用以下命令来卸载 Node.js 模块。

npm uninstall <Module Name>

卸载后，你可以到 /node\_modules/ 目录下查看包是否还存在，或者使用以下命令查看

npm ls

## 更新模块

我们可以使用以下命令更新模块：

npm update <Module Name>

## 搜索模块

使用以下来搜索模块：

npm search <Module Name>

参考文章：http://www.runoob.com/nodejs/nodejs-npm.html