

Defining an Application and Managing Dependencies



Paul O'Fallon

@paulofallon



Overview



Modularizing a Node application

Application dependencies

Introduce the npm CLI and repository



Modularizing a Node.js Application

waitress.js

```
const cook = require('./cook')
const Customer = require('./customer')
// cook.prepareFood()
// new Customer()
```

cook.js

```
const ingredients = 'stuff'
const prepareFood = (order, done) => {
  // prepare the food
}
module.exports = { prepareFood }
```

customer.js

```
const moment = require('moment')
class Customer {
  // methods and properties
}
module.exports = Customer
```



What Is npm?



A Package Repository



A Command Line Application

Initializing a Project

package.json

```
{
  "name": "waitress-sync",
  "version": "1.0.0",
  "description": "",
  "main": "waitress.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "repository": {
    "type": "git",
    "url": "git+https://github.com/pofallon/nodejs-big-picture.git"
  },
  "author": "",
  "license": "MIT",
  "bugs": {
    "url": "https://github.com/pofallon/nodejs-big-picture/issues"
  },
  "homepage": "https://github.com/pofallon/nodejs-big-picture#readme"
}
```

\$ npm init



Adding Dependencies

package.json

```
{  
  // ...  
  "repository": {  
    "type": "git",  
    "url": "git+https://github.com/pofallon/nodejs-big-picture.git"  
  },  
  "author": "",  
  "license": "MIT",  
  "bugs": {  
    "url": "https://github.com/pofallon/nodejs-big-picture/issues"  
  },  
  "homepage": "https://github.com/pofallon/nodejs-big-picture#readme",  
  "dependencies": {  
    "moment": "^2.24.0"  
  }  
}
```

```
$ npm install --save moment
```



Adding Development Dependencies

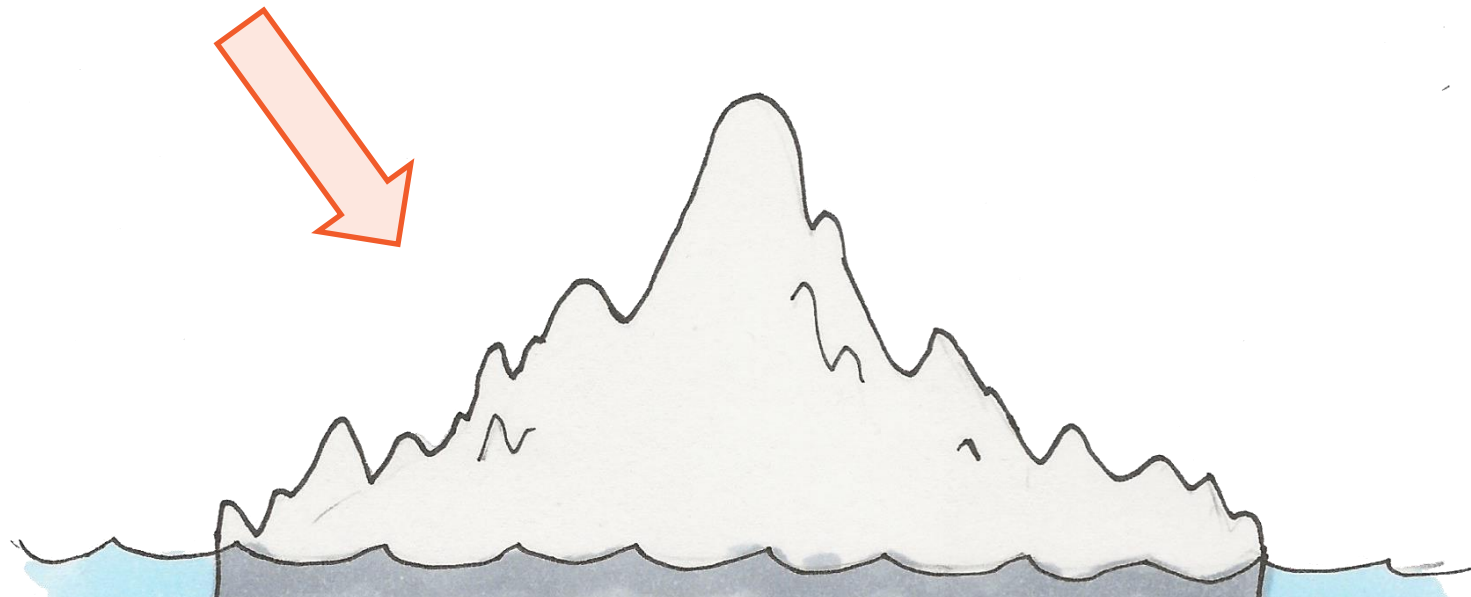
package.json

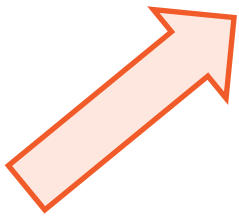
```
{  
  // ...  
  "author": "",  
  "license": "MIT",  
  "bugs": {  
    "url": "https://github.com/pofallon/nodejs-big-picture/issues"  
  },  
  "homepage": "https://github.com/pofallon/nodejs-big-picture#readme"  
  "dependencies": {  
    "moment": "^2.24.0"  
  },  
  "devDependencies": {  
    "mocha": "^6.0.2"  
  }  
}
```

```
$ npm install --save-dev mocha
```



Modules we required





**All our project's
dependencies**



What Was Actually Installed?

package.json

```
{
  // ...
  "dependencies": {
    "moment": "^2.24.0"
  },
  "devDependencies": {
    "mocha": "^6.0.2"
  }
}
```

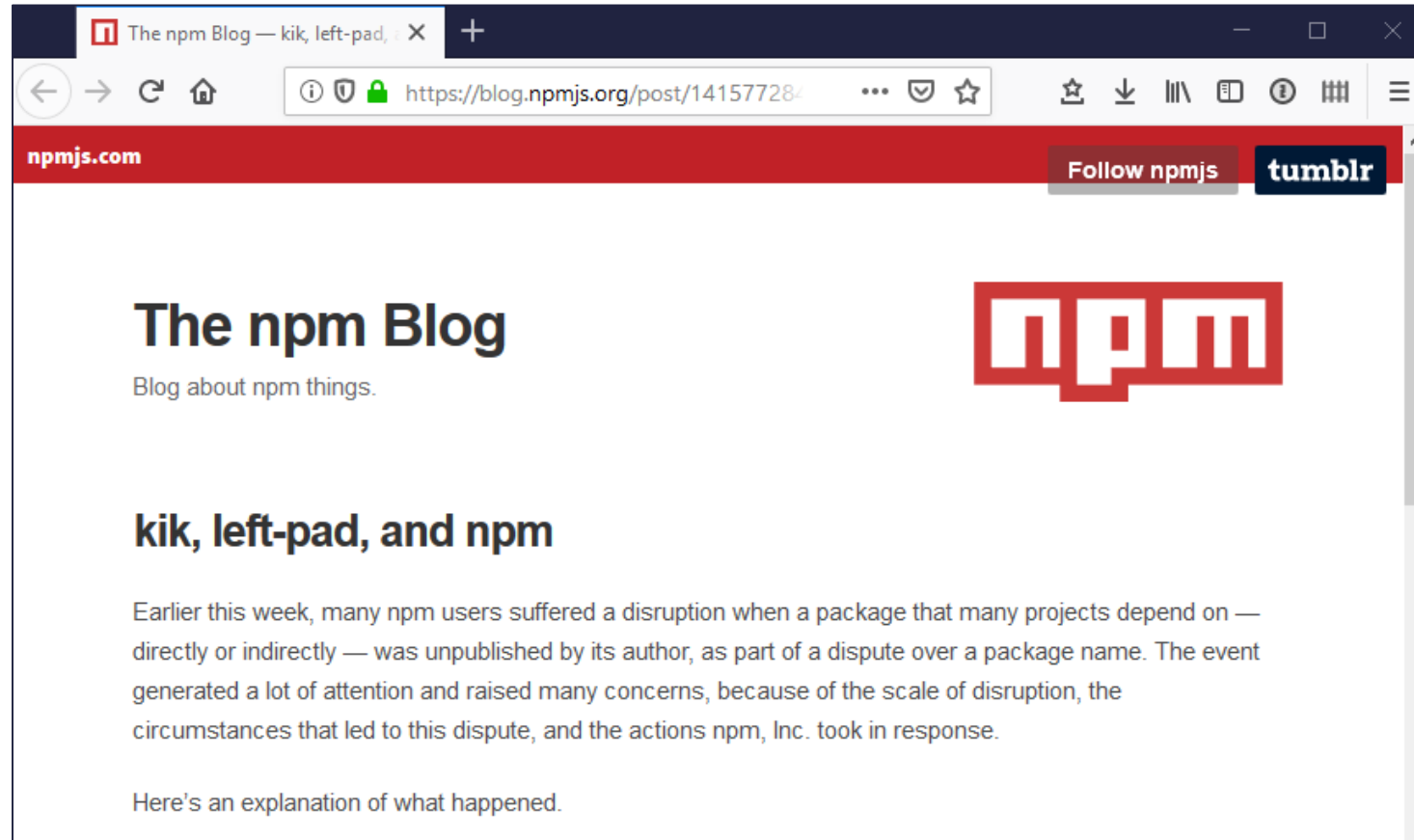
```
$ ls node_modules
ansi-colors  decode-uri-component  has  js-yaml  once  snapdragon
ansi-regex  define-properties  has-flag  kind-of  os-locale  snapdragon-node
ansi-styles  define-property  has-symbols  lcid  p-defer  snapdragon-util
argparse  detect-file  has-value  locate-path  p-finally  source-map
arr-diff  diff  has-values  lodash  p-is-promise  source-map-resolve
arr-flatten  end-of-stream  he  log-symbols  p-limit  source-map-url
arr-union  es-abstract  homedir-polyfill  map-age-cleaner  p-locate  split-string
array-unique  es-to-primitive  inflight  map-cache  p-try  sprintf-js
assign-symbols  escape-string-regexp  inherits  map-visit  parse-passwd  static-extend
atob  esprima  ini  mem  pascalcase  string-width
balanced-match  execa  invert-kv  micromatch  path-exists  strip-ansi
base  expand-brackets  is-accessor-descriptor  mimic-fr  path-is-absolute  strip-eof
brace-expansion  expand-tilde  is-buffer  minimatch  path-key  strip-json-comments
braces  extend-shallow  is-callable  minimalist  posix-character-classes  supports-color
browser-stdout  extglob  is-data-descriptor  mixin-deep  pump  to-object-path
cache-base  fill-range  is-date-object  mkdirp  regex-not  to-regex
camelcase  find-up  is-descriptor  mocha  repeat-element  to-regex-range
chalk  findup-sync  is-extendable  moment  repeat-string  union-value
class-utils  flat  is-extglob  ms  require-directory  unset-value
cliui  for-in  is-fullwidth-code-point  nanomatch  require-main-filename  unix
code-point-at  fragment-cache  is-glob  nice-try  resolve-dir  use
collection-visit  fs.realpath  is-number  node-environment-flags  resolve-url  which
color-convert  function-bind  is-plain-object  npm-run-path  ret  which-module
color-name  get-caller-file  is-regex  number-is-nan  safe-regex  wide-align
component-emitter  get-stream  is-stream  object-copy  semver  wrap-ansi
concat-map  get-value  is-symbol  object-keys  set-blocking  wrappy
copy-descriptor  glob  is-windows  object-visit  set-value  y18n
cross-spawn  glob-all  isarray  object.assign  shebang-command  yargs
debug  global-prefix  isexe  object.getownpropertydescriptors  shebang-regex  yargs-parser
decamelize  growl  isobject  object.pick  signal-exit  yargs-unparser
$
```

16Mb in 180 modules



Lots of Modules with Many Dependencies

What could possibly go wrong?



<https://blog.npmjs.org/post/141577284765/kik-left-pad-and-npm>



Yarn as an npm CLI Alternative

