

Gatsby's Node.js support policy

Gatsby aims to support any version of Node that has a release status of *Current*, *Active*, or *Maintenance*. Once a major version of Node reaches *End of Life* status Gatsby will stop supporting that version.

Gatsby will stop supporting the *End of Life* Node release in a minor version.

Check [Node's releases document](#) for version statuses.

What version of Node.js do I have?

Run `node -v` in a terminal to see which version of Node.js you have.

```
node -v
v12.21.0
```

This example shows Node.js version 12, specifically v12.21.0.

Upgrading from Node.js version 10

Node.js version 10 *End-of-life* date is April 30, 2021. Many of Gatsby's dependencies are updating to Node.js version 12 and above. Gatsby must also update in order to deliver new features and bug fixes more quickly.

Generally, it's recommended to use [the Node version whose status is Active LTS](#) (Node 14 at time of writing).

What about Node.js 13? Stable versions of Node.js are evenly numbered releases - Node.js 8, Node.js 10, Node.js 12 etc. Only use uneven release numbers if you'd like to try cutting-edge and experimental features.

There are multiple ways to update your version of Node.js depending on how you originally installed it. Read on to find the best approach for you.

Using Homebrew

This is the recommended way to install a newer version of Node.

You will have Homebrew installed on your computer if you [followed part zero of the Gatsby tutorial](#). Homebrew is a program that allows you to install specific versions of Node.js (and other software).

To update from Node.js 10 to Node.js 12 using Homebrew, open a terminal and run the following commands:

```
brew search node
```

You should see output similar to this:

```
brew search node
==> Formulae
heroku/brew/heroku-node ✓      llnode      node@12
nodebrew
leafnode                      node ✓      node@10
nodeenv
libbitcoin-node              node-build  node_exporter
nodenv
```

You're interested in the next stable version of Node.js after Node.js 10, which is Node.js 12. Homebrew makes this available in a package called `node@12`. Run:

```
brew install node@12
```

Once that's complete, run:

```
node -v
```

to confirm that you've upgraded from Node.js version 10 up to version 12.

Using a Node.js version management package

There are two popular packages used for managing multiple versions of Node.js on your system. Use one of these to update to a newer version of Node.js if they're already available on your computer.

These packages are very useful for people that regularly work with different versions of Node.js.

nvm

Run

```
nvm
```

in a terminal to see if nvm is installed on your system. If it's installed, you can run:

```
nvm install 12  
nvm alias default 12
```

to install and use Node.js version 12.

[Check nvm's documentation for further instructions.](#)

n

Run:

```
n
```

in a terminal to see if n is installed on your system. If it's installed, you can run `n 12` to install and use Node.js version 12.

[Check n's documentation for further instructions.](#)

Installing from nodejs.org

If you aren't using any of the previously listed installation methods, you can [download a Node.js installer directly from nodejs.org](#).

Gatsby's recommended way to install Node.js is by using Homebrew. Refer to the previous [Homebrew section of this document](#) for more info.

Conclusion

Gatsby takes backwards compatibility seriously and aims to support older versions of Node.js for as long as possible. We understand that juggling different software versions is not a productive way to spend your day.

Gatsby also relies on a huge ecosystem of JavaScript dependencies. As the ecosystem moves away from older, unsupported Node.js versions we have to keep pace to ensure that bugs can be fixed and new features can be released.

In this document, you learned how you upgrade from Node.js version 10 to Node.js version 12.