

This page describes how to compile three.js into a compressed JavaScript file.

Quickstart

- Install Node.js.
- Clone repository, use `-depth` parameter to prevent fetching all commit history. Also you can just download and unzip folder.

```
git clone --depth=30 https://github.com/mrdoob/three.js.git
```

- Go into the three.js directory.

```
cd ./three.js
```

- Install build dependencies

```
npm i
```

- Run the build script.

```
npm run build
```

The ESM build file is at `build/three.module.js`, with `build/three.js` being the UMD version and `build/three.min.js` the compressed UMD version.

When developing it is recommended to use `npm start` or `npm run dev`. This will host a local web server and watch the files in `src` folder and automatically perform a build when source files are changed.

Why compression?

The source code of Three.js is deliberately easy to read. This means that it's split into several files and classes. While that's great for developing and hacking on Three.js, it's not that great when deploying code to the production server.

In production, you want to - use the least amount of files possible (to minimize the number of connections to your server) - transmit as few bytes as possible (to save on bandwidth and on wait-time on both sides)

So how do we put all files into just one and make it smaller than the sum of the parts? Well, the answer is the awesome combination of our build script plus JavaScript compressors!