
nodist

[gitter](#) [join chat](#)

A node.js and npm version manager for the windows folks out there. Inspired by n. And nodenv.

Heads up! Nodist v0.10 is here! Nodist supports cmd, Powershell, Git bash and Cygwin!

```
1 C:\> nodist + 5
2 5.11.0
3
4 C:\> nodist global 5
5 5
6
7 C:\> node -v
8 v5.11.0
9
10 C:\> nodist
11 0.10.26
12 4.4.3
13 > 5.11.0 (global: 5)
```

(see Usage)

Known issues

Please read this!

Over the past months several problems have presented themselves, which are due to the way nodist works and are hard to fix:

- process signals: Nodist employs a wrapper executable to shim the functionality of Node. Since Windows doesn't have signals, sending a SIGTERM, or similar will probably not be propagated to the actual node process, but get stuck in the shim. (see #173)
- native modules: Since the node version changes at the mercy of the shim executable, based on env vars, target directory and the global setting, and availability of node versions, it is possible that locally or globally installed node modules that depend on a specific version of node (usually native modules and downloaders) stop working. [npm rebuild](#) makes things work again in these cases.

Installation

Nodist was designed to replace any existing node.js installation, so *if node is already installed on your machine, uninstall it first.*

with the installer

1. Download the installer from the releases page
2. Run the installer and follow the install wizard

via chocolatey

For this you'll need chocolatey, of course.

1. `choco install nodist`

Activating nodist in git bash

Git bash integration is setup automatically.

Activating nodist in cygwin

Before you are able to use nodist in cygwin you need to run the following in your cygwin terminal after installing nodist:

```
1 cat "$NODIST_PREFIX/bin/nodist_bash_profile_content.sh" >> ~/.bashrc
2 source ~/.bashrc
```

Activating nodist in PowerShell

You might need to 'Unblock' the file `bin\nodist.ps1` by right-clicking on it in the Explorer and selecting that menu entry.

If you cannot see the unblock option after right-clicking > Properties in Explorer you can also perform the unblock via the following PowerShell command:

```
1 unblock-file -path "C:\Program Files (x86)\Nodist\bin\nodist.ps1"
```

If you still cannot run nodist you may also need amend your ExecutionPolicy setting.

Warning: Please ensure you understand the risks associated with the different execution policies by first reviewing [about_Execution_Policies](#) - Microsoft Docs

```
1 Set-ExecutionPolicy -ExecutionPolicy RemoteSigned
```

Make it work in your IDE

If your IDE cannot access node or npm right away, don't fret! Find the relevant configuration settings and either set `path\to\Nodist\bin` as the Node installation dir or set `path\to\Nodist\bin\node.exe` as the path to the node binary directly.

Similar for npm: either set `...\Nodist\bin` as the installation path or `...\Nodist\bin\npm.cmd` directly. Some IDEs (eg. WebStorm) require you to set the path to an npm package containing `bin\npm-cli.js`. Nodist comes with a proxy npm folder: Simply set `...\Nodist\npm` as the npm package (this feature is experimental and feedback is appreciated!).

Migrating from 0.7 or lower

If you're looking to upgrade your Nodist installation, the easiest way is to uninstall (see below) the old installation and install the new version with the installer above. You may need to reset your per-directory node version settings in order for them to work in v0.8 (ones set using nodist v0.6 should work fine, ones set using v0.7 will not).

Important usage changes in v0.8 Starting in v0.8 Nodist employs lazy version pattern evaluation. This means that setting versions per env/locally/globally doesn't set an explicit version, if you didn't give one. Instead the node.exe shim chooses a suitable version *at runtime*. Commands for installing a version and setting a requirement have been separated, thus, to update your node version (if your global version is set to 6, e.g.), you now need to run `nodist + 6` (i.e. `nodist 6` doesn't do that for you anymore), which is probably how it should have worked all along.

Uninstall (v0.7)

1. Run the uninstaller either from the directory where you installed nodist, or from the Software Control Panel
2. Make sure to completely remove the nodist directory. (This will remove all your globally installed npm modules. If you don't want that, remove everything else, other than `nodist\bin\.`)

Uninstall (before v0.7)

1. Remove `<...path...>\nodist\bin` from your path. (how?).
2. Remove the `NODIST_PREFIX` env var.

-
3. Delete the nodist directory
 4. Run `npm config delete prefix` on the command prompt to restore NPM functionality if you are going back to the official node install.

Usage

Nodist allows you to set version requirements for different scopes. Version requirements can be fully specified versions, like `4.0.0` or patterns like `0.12`, `4.x`, `~5` or `latest`. `io.js` is supported natively: Since node and `io.js` versions form a continuum you can simply use `io.js` versions as if they were node versions. Setting a requirement installs a matching version only if there is no other matching version already installed; otherwise existing installed versions will get referenced when executing `node.exe`.

Scope precedence

The following is a list of all scopes ordered by precedence (the first scope is the one with the highest priority; only if it's not set, the second scope is examined).

1. Environment (`NODIST_NODE_VERSION` and `NODIST_NPM_VERSION` env vars)
2. (optional:) Package (`package.json` with an `engines` field in the *directory of interest* or one of its parent directories)
3. Directory (`.node-version` or `.npm-version` in the *directory of interest* or one of its parent directories)
4. Global (globally set node or npm version)

When you're just running node, the *directory of interest* is the directory of the javascript file to be executed. When running npm, it is the current working directory.

Package.json inspection is turned off by default as of nodist v0.8.5. You can turn it on by setting `NODIST_INSPECT_PACKAGEJSON=1`.

npm / npx

Any instances of node invoked by npm/npx will be locked to the same version npm/npx runs on.

Currently, all node and npm versions share the same global npm module space.

If you have installed native modules (globally or locally) you may have to run `npm rebuild` after changing the node version (implicitly or explicitly). There is an open issue about how to avoid rebuilding globally installed native modules, feedback/input is welcome.

The npx shim will try to run the npx shipped with the currently selected npm version.

Commands

All commands automatically install the latest matching version before setting the version pattern.

```
1 > nodist
2 # Lists installed versions highlighting the active ones.
```

```
1 > nodist global 4.x
2 # Sets the global node version requirement
```

```
1 > nodist local 4.x
2 # Sets the node version requirement per directory (including all
   subdirectories).
```

```
1 > nodist env 4.x
2 # Sets the node version requirement per terminal.
```

```
1 > nodist npm global 3.x
2 # Set global npm version requirement.
3
4 > nodist npm global match
5 # Tell nodist to always choose the npm version that matches the current
   node version.
6 # (the current node version may be determined by env, local or global
   requirements)
```

```
1 > nodist npm local 2.x
2 # Set the npm version requirement for the current directory.
```

```
1 > nodist npm env 2.x
2 # Set the npm version requirement for the current terminal environment.
```

```
1 call nodist env 4.x
2 # In a batch script use `call`.
```

```
1 > nodist dist
2 # Lists all available node versions.
```

```
1 > nodist + 4.x
2 # Just checks, if the version is installed and installs it if not.
3
4 > nodist + all
5 # will install *everything*.
```

```
1 > nodist - 4.1.1
```

```
2 # Removes a version.
```

```
1 > nodist --help
2 # Displays a complete list of commands with examples.
```

Settings

```
1 > set HTTP_PROXY=http://myproxy.com:8213
2 # Set a proxy to use for fetching the executables
3 # (you may also use `HTTP_PROXY`/`http_proxy`/`HTTPS_PROXY`/`https_proxy`).
```

```
1 > set NODIST_NODE_MIRROR=https://mymirror.com/dist
2 # Set a mirror that has the same directory structure as https://nodejs.org/dist
3 # (you may also use `NODIST_IOJS_MIRROR` in the same fashion).
```

```
1 > set NODIST_X64=0
2 # (Set to `1` to enforce 64bit, `0` to enforce 32bit.)
```

```
1 > set NODIST_INSPECT_PACKAGEJSON=1
2 # Enable package.json inspection: Nodist will check the engines field of the package.json file of scripts to be executed and use the node version specified there. Turned off by default as of v0.8.5.
```

```
1 > set NODIST_GITHUB_TOKEN=your_github_personal_access_token
2 # By default Nodist makes request to the GitHub API to list npm releases without any authentication, which is subject to [rate limiting](https://docs.github.com/en/rest/overview/resources-in-the-rest-api#rate-limiting). By using your own [Personal Access Token](https://docs.github.com/en/authentication/keeping-your-account-and-data-secure/creating-a-personal-access-token) Nodist will be able to make more requests per hour.
3 # Note: for security purposes it's best to use a token just for Nodist. The `repo:public_repo` scope should be more than sufficient.
```

Details

Node executables are stored in `NODIST_PREFIX\v` and `NODIST_PREFIX\v-x64`, npm versions in `NODIST_PREFIX\npmv`. The `globalnode.exe` is a shim and chooses the right node version to run based on the various version settings. The same applies for npm.

As the global node version will be subject to change, `nodist` comes with its own dedicated node binary.

Got ideas? Doesn't work for you? Want to give feedback?

File an issue and tell me what you'd change or add or what doesn't work for you. Every issue is welcome!

Malware warnings

Nodist is not a virus. Nonetheless, some malware detection tools report the Nodist installer as a virus. If this happens, please contact the vendor of your malware detection tool and report the Nodist installer as a false positive.

Debugging

To see all debug messages, set the `DEBUG` env var before running `nodist` or `node` as follows:

```
1 > set DEBUG=nodist:*
```

Testing

The default test suite can be run using `npm`

```
1 $ npm test
```

Testing also accepts env variables for using a mirror to download from, as well as setting a proxy.

```
1 > set NODIST_NODE_MIRROR=http://nodejs.serverpals.com/dist
2 > vows --spec test\cli-test.js
```

Building

Building `nodist` requires * `go` for compiling the shim * `NSIS v3` for compiling the installer * `NSIS` plugins will be automatically downloaded before build: * `NSIS Plugin: AccessControl` * `NSIS Plugin: EnVar Plugin` * `node.js` for running the build script * and `npm` for installing `nodist`'s dependencies

If you have met all requirements, run the build command:

```
1 > npm run build
```

Afterwards you'll find the installer in `build/out/NodistSetup-vX.X.X.exe` and fully prepared installation folder in `build/out/staging` (you could zip this, for example). The chocolatey

package will be in `build/out/package`, you can run `cpack` and `cpush --source https://chocolatey.org/` inside that directory (if you are a registered maintainer).

Legal

Copyright (c) 2012-2023 by Marcel Klehr, Bryan Tong (@nullivex)
MIT License

Changelog

v0.10.3 * Fix installing of npm versions

v0.10.2 * Fix building shims (for newer go versions) by using go modules * Fix npm shim to use correct node version * Add npx shim (works only for npm versions that ship with npx) * Fix getting latest npm version * Use last available x86 version for building (first node 18 versions are not available for x86) * Resolve symlinks for npm's node_modules to support npm >= 8 * Use octokit lib to access github, add NODIST_GITHUB_TOKEN env variable PR#246 * Fix `npm ls` to resolve correct version PR#240 * Minimal node version to build nodist raised to 12

v0.9.1 * Fix issue with deprecated call to Tar.Extract in the NPM handler.

v0.9.0

- Maintenance release. First release by @nullivex
- Update to latest dependencies.
- Fix for installing NPM newer than 6.1.0 PR#222
- Add additional Nodist Powershell Activation PR#213
- Fix `npm ls` to resolve correct version PR#210

v0.8.8

- Fix input validation for setting global version requirement
- Fix npm prefix not being set correctly

v0.8.7

- Update dependency: request (had a vulnerability)
- Fix shim: Correctly proxy signals
- Simplify usage and docs

v0.8.6 * Fix installer: Increase MAX_LEN for manipulating PATH to 8000 * Add support for WebStorm and VisualStudio

v0.8.5 * Disable package.json inspection by default

v0.8.4 * Fix version spec parsing in shims

v0.8.3 * Fix npm builds: Lock node version for npm's child processes

v0.8.2 * Fix Git bash support and add support for Cygwin * Fix `nodist dist` output being unsorted

v0.8.1 * Fix usage/help output: `stable` has been removed * Fix installer: Use the correct path to npm when setting npm prefix

v0.8.0 * Add NPM version management (thanks to @nullivex) * Treat io.js versions as node versions * Allow setting ranges in global/local/env (don't resolve before setting versions) * Drop support for setting node command line args * Respect engines field declaration in package.json * Fix local switching: Use the target script's dir as the base dir * Allow setting env vars for mirror support * Support bash * Remove selfupdate command * Fix help flag * [installer] Fix: Set system not user PATH * [installer] Fix: auto-detect x64 arch * Improve build script * Revive chocolatey package

v0.7.2 * correct version of NPM * Fixed offline support (#104)

v0.7.1 * Fix x64 support

v0.7.0 (thanks to @nullivex) * Add support for hashing downloading binaries and comparing to upstream * Code refactor to implement standards similar to npmjs style * Implement download progress bar * Add debugging output to aid in development and issue handling * Bump NPM version to 3.3.8 * Improve testing and offer ability to see test output * Testing now accepts mirror env variables * Env variables can now be used to point nodist at a mirror * Only downloads binaries from HTTPS now as node is phasing out HTTP support

v0.6.1 * Fix for node v4 dist directory structure (thanks to @jakub-g)

v0.6.0 * Write out version spec that's being used, to make version errors more intuitive. (thanks to @blzaugg) * Fix replace for nodistPrefix and envVersion (thanks to @sdovenor) * Fix chocolatey package (thanks to @u9520107) * Add support for io.js

v0.5.2 * Fix nodist version

v0.5.1 * Fix npm prefix in `nodist selfupdate`

v0.5.0 * Use a binary shim * Support for localized version switching

v0.4.8 * Check in dependencies to avoid relying on npm on install

v0.4.7 * Update bundled npm to v1.4.6 and check-in all files

v0.4.6 * Update pre-packaged npm to v1.4.3

v0.4.5 * Fix `nodist -v` in powershell (thanks to @devert for reporting)

v0.4.4 * Fix GOTO in batch file (thanks to @BergWerkGIS)

v0.4.3 * Fix a type coercion catch in x64 detection (NODIST_X64=0 had the same effect as NODIST_X64=1; thanks to @springmeyer for reporting)

v0.4.2 * Use the right URI for fetching the x64 executables

v0.4.1 * x64 auto-detection

v0.4.0 * Refactor 64bit support (All commands now work with x64 versions) * Make npm use the user-chosen node version (thanks to @k-j-kleist for reporting) * Fix some minor bugs in error handling * Don't try installing 'vUpdate' after running nodist update * Install latest stable version after running nodist update

v0.3.11

- Fix 'use' command in powershell (thanks to @paulbatum for reporting)

v0.3.10

- Fix 'use' by fixing 'path' and 'bin' commands

v0.3.9

- Fix renaming mistake (thanks to @paulbatum for reporting)
- Tighten `nodist update`

v0.3.8

- Add support for x64 versions (thanks to @CycloPH)
- Improve performance by caching version lists

v0.3.7

- Fix `nodist run` and `list` commands
- Allow people to use an http proxy (thanks to @gratex)

v0.3.5

- Fix install script. How did that go astray?

v0.3.4

- Fix error message if `n.checkout` fails

v0.3.3

- Fix installer (thanks to @Ciantic for reporting)

-
- Fix a bug introduced by npm update

v0.3.2

- Update npm on install/update

v0.3.1

- Optimized `nodist + all`
- Use a clean npm
- Don't bundle dependencies, anymore – `nodist update` must be run at install

v0.3.0

- Dropped `optimist` (caused more problems than it solved)
- Using `node-semver`, now, for better usability... (fixes #10)
- Allow use of version patterns for every command (fixes #9)
- Fix issue #8: Version patterns should *require* internet access

v0.2.8

- Fix use command

v0.2.7

- Deploy by copying node.exe, again, this should allow people to use nodist together with node-mon

v0.2.6

- Fix a bug, that used to break everything on Win7 x64

v0.2.5

- Updated npm to v1.1.48

v0.2.4

- Fix `nodist update` command to work from all drives
- Add basic support for MinGW shell (thanks to jdiamond) (update and use commands are still missing)
- Fix tests
- Allow npm updates with `npm u npm -g`
- updated npm

v0.2.3

-
- Install latest stable as default version
 - Add update command
 - Updated npm