Volta Node toolchain manager

CONCEPTS

Volta swaps tool versions per directory: a pin in package.json wins, otherwise your default applies. No shell hook or use command needed.

node / npm / tsc you type a command
↓ shim on PATH
Volta shim in ~/.volta/bin
↓ looks up
package.json "volta" else your default
↓ runs
Pinned binary right Node engine

Global packages are pinned to the Node they were installed with and keep it until you reinstall them.

Commit the "volta" block in package.json so the whole team and CI get the same toolchain.

INSTALL VOLTA

curl https://get.volta.sh | bash
Install on Linux, macOS, WSL

winget install Volta.Volta
Install on Windows

volta setup
Add shims to PATH for your user

volta --version
Show Volta version

volta completions *<shell>*
Print shell completions

Manual setup: export VOLTA_HOME=$HOME/.volta and put $VOLTA_HOME/bin first on PATH.

DEFAULT TOOLS

volta install node
Default to the latest LTS Node

volta install node@<*version*>
Default to an exact or partial version

volta install npm@<*version*>
Default npm instead of the bundled one

volta install yarn@<*version*>
Default Yarn

DEFAULT TOOLS (cont.)

volta install *<package>*
Install a global package binary

volta uninstall *<package>*
Remove a global package or tool

volta fetch *<tool>*@*<version>*
Download to cache only, no default

install changes your default everywhere outside pinned projects; use pin for a single project.

PIN PER PROJECT

volta pin node@<*version*>
Pin Node in package.json

volta pin node@lts
Pin the current LTS Node

volta pin npm@<*version*>
Pin npm

volta pin yarn@<*range*>
Pin Yarn

What pin writes

```
"volta": {
  "node": "20.16.0",
  "npm": "10.8.1"
}
```

pin only handles Node and package managers. Change library versions with npm install or yarn add.

INSPECT

volta list
Show the active toolchain

volta list all
Everything installed

volta list node
Installed Node versions

volta list --default
Show your defaults only

volta which *<binary>*
Real path behind a shim

RUN ONCE

volta run --node *<version> <cmd>*
Run with another Node