Java Forum Stuttgart, 10.07.2025

Dev Env Reloaded
- Die lokale Entwicklungsumgebung neu gedacht

Sandra Parsick
@sparsick@mastodon.social
mail@sandra-parsick.de

Wer bin ich?

- Sandra Parsick
- Freiberuflicher Softwareentwickler und Consultant im Java-Umfeld
- Schwerpunkte:
  - Java Enterprise Anwendungen
  - Agile Methoden
  - Software Craftmanship
  - Automatisierung von Entwicklungsprozessen
- Trainings
- Workshops

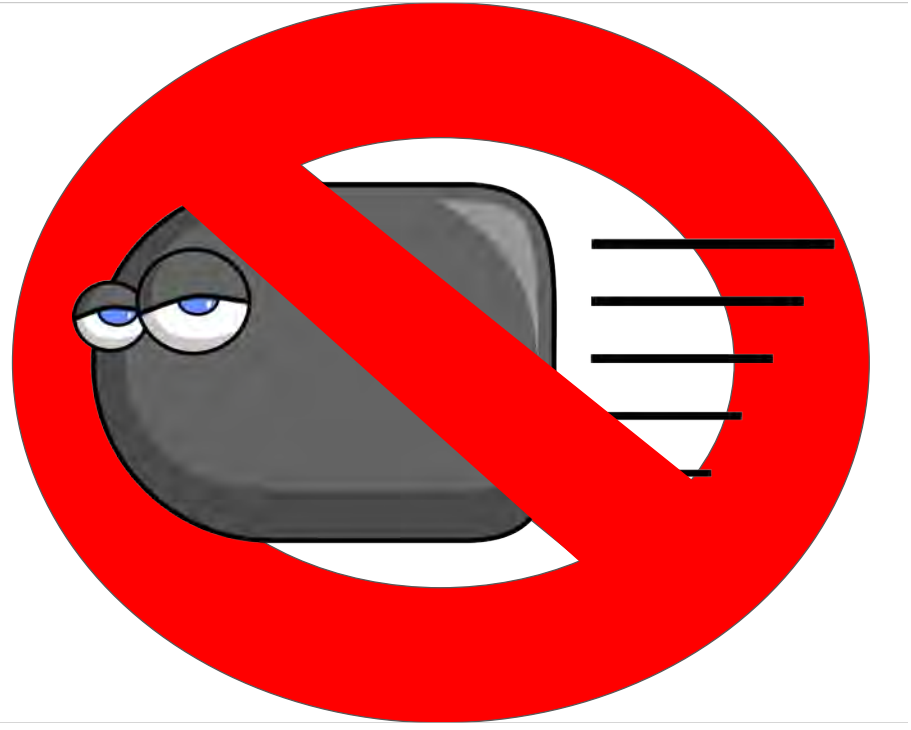
mail@sandra-parsick.de

@sparsick@mastodon.social

https://www.sandra-parsick.de

https://ready-for-review.dev

Disclaimer




Ich habe einen neuen Rechner
Kann mir jemand helfen den neuen Tech Stack aufzusetzen
Das dauert was länger. Ich habe Umrüstzeiten

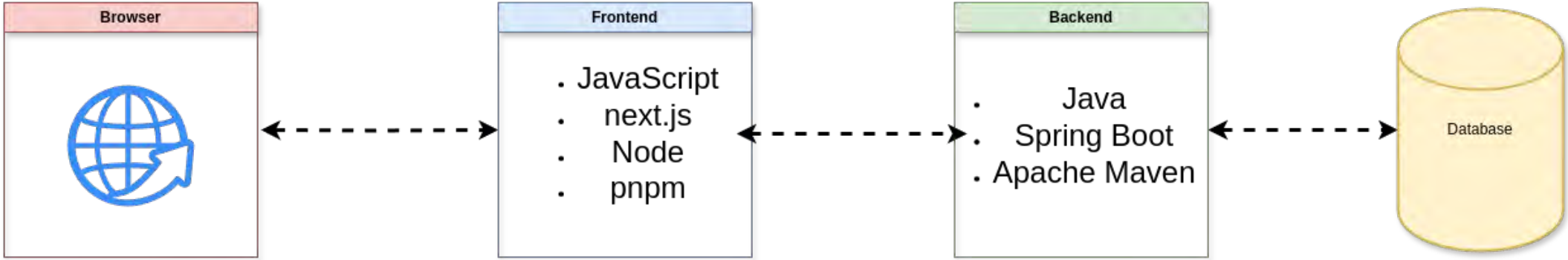
Lösungsansätze

(1) Kombination aus verschiedenen Hilfsmitteln

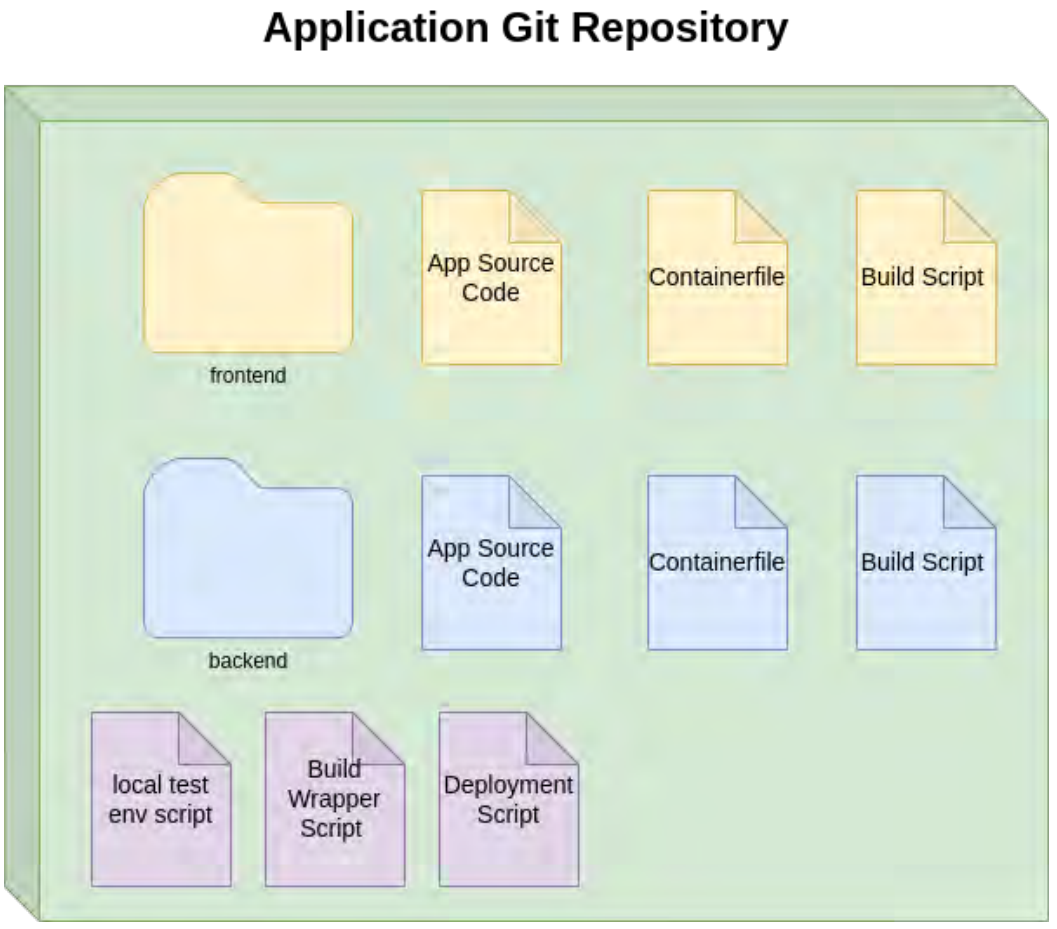
(2) Container-basiert

(3) Reproduzierbare Umgebung ohne Container

Beispiel-Anwendung



Tool Stack



JS

node.js

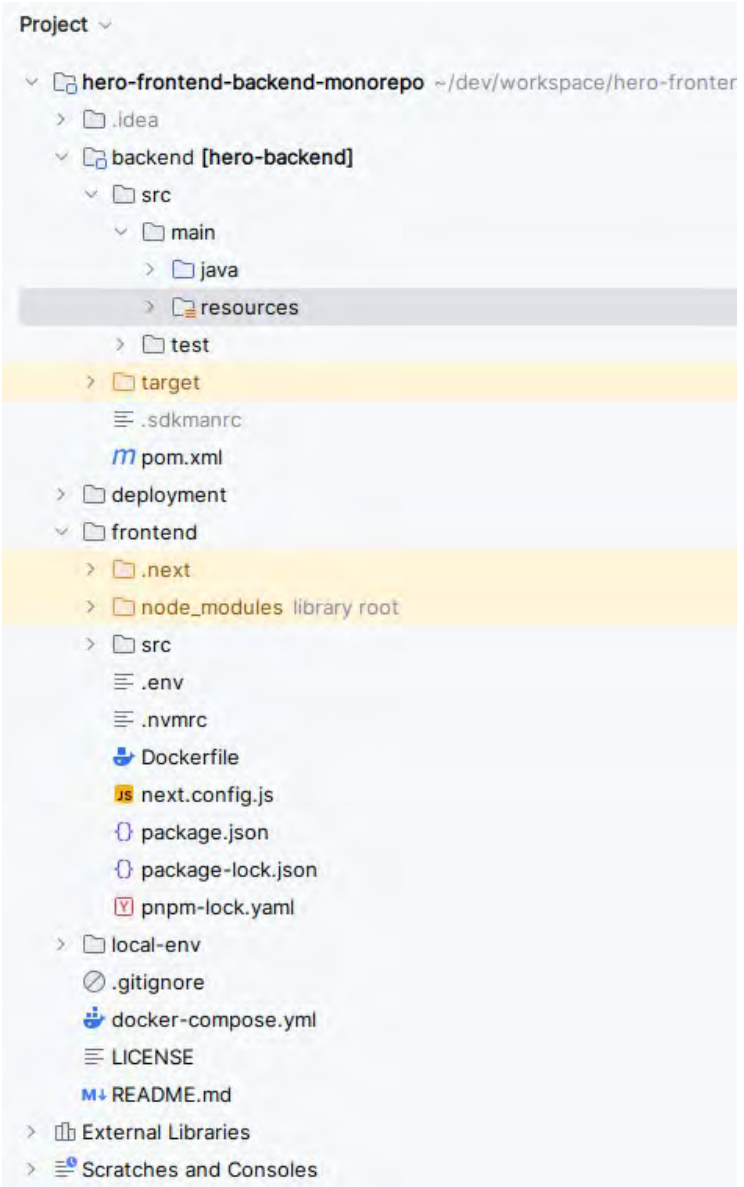
pnpm

NEXT.JS

spring boot

Java

Maven


Project
hero-frontend-backend-monorepo ~/dev/workspace/hero-fronter
.idea
backend [hero-backend]
src
main
java
resources
test
target
.sdkmanrc
pom.xml
deployment
frontend
.next
node_modules library root
src
.env
.nvmrc
Dockerfile
next.config.js
package.json
package-lock.json
pnpm-lock.yaml
local-env
.gitignore
docker-compose.yml
LICENSE
README.md
External Libraries
Scratches and Consoles


eclipse


Vim


IJ

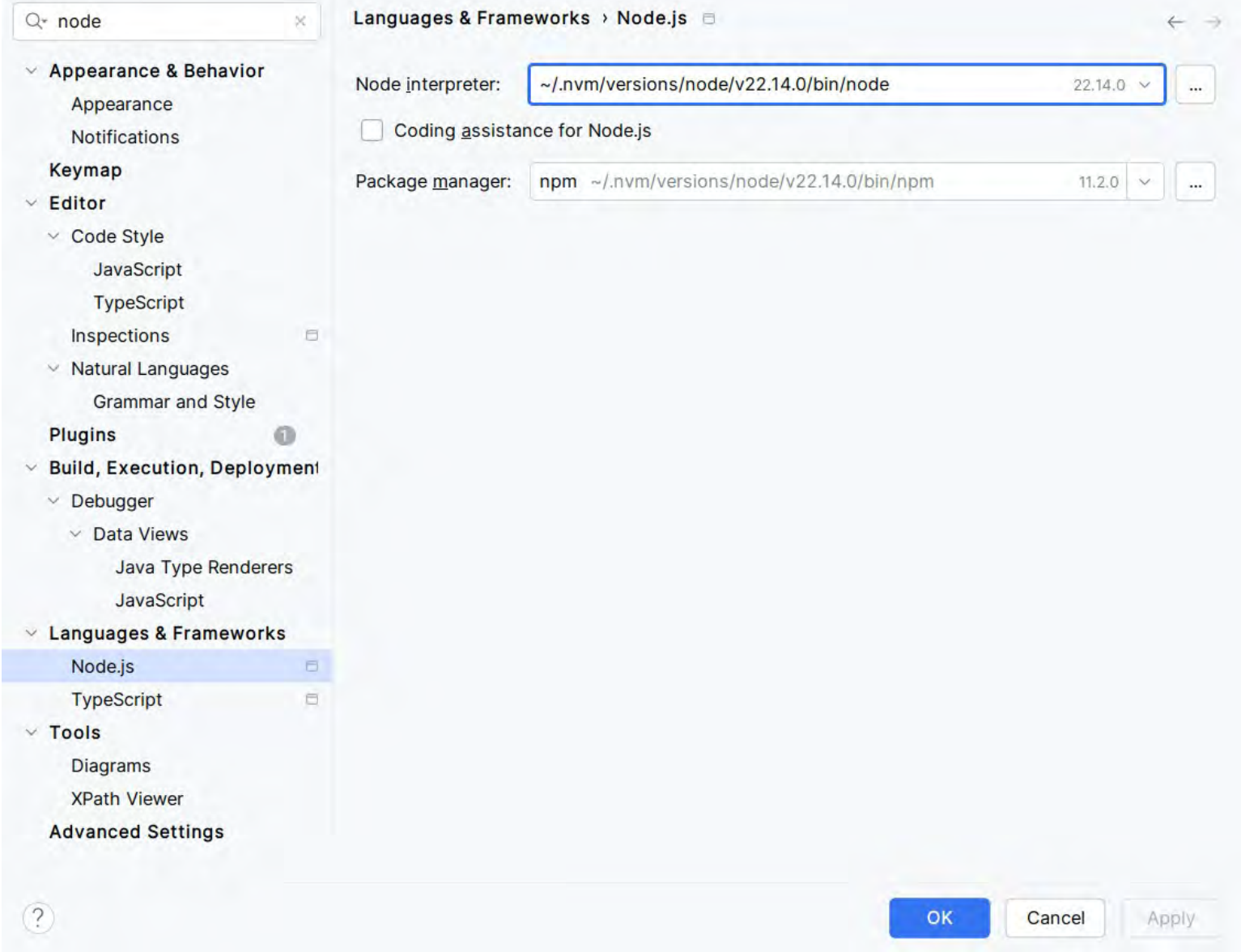
Lösungansatz: Kombination aus verschiedenen Hilfsmitteln


Installation und
Verwaltung der SDKs

Frontend-Entwicklung

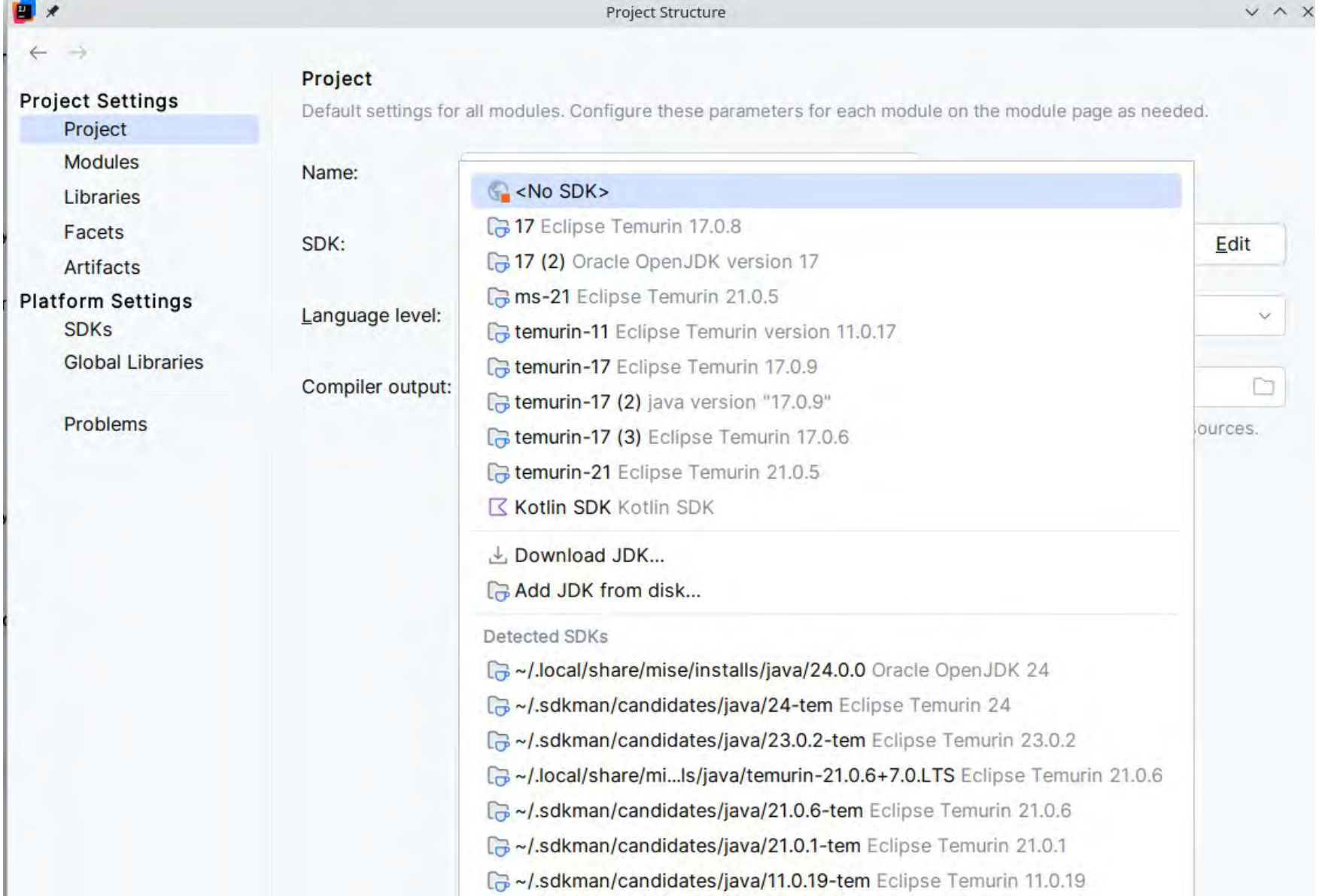
Node Version Manager

Demo


node
Appearance & Behavior
Appearance
Notifications
Keymap
Editor
Code Style
JavaScript
TypeScript
Inspections
Natural Languages
Grammar and Style
Plugins
Build, Execution, Deployment
Debugger
Data Views
Java Type Renderers
JavaScript
Languages & Frameworks
Node.js
TypeScript
Tools
Diagrams
XPath Viewer
Advanced Settings
Languages & Frameworks › Node.js
Node interpreter:
~/.nvm/versions/node/v22.14.0/bin/node
22.14.0
...
Coding assistance for Node.js
Package manager:
npm ~/.nvm/versions/node/v22.14.0/bin/npm
11.2.0
...
OK
Cancel
Apply

Backend-Entwicklung

Demo

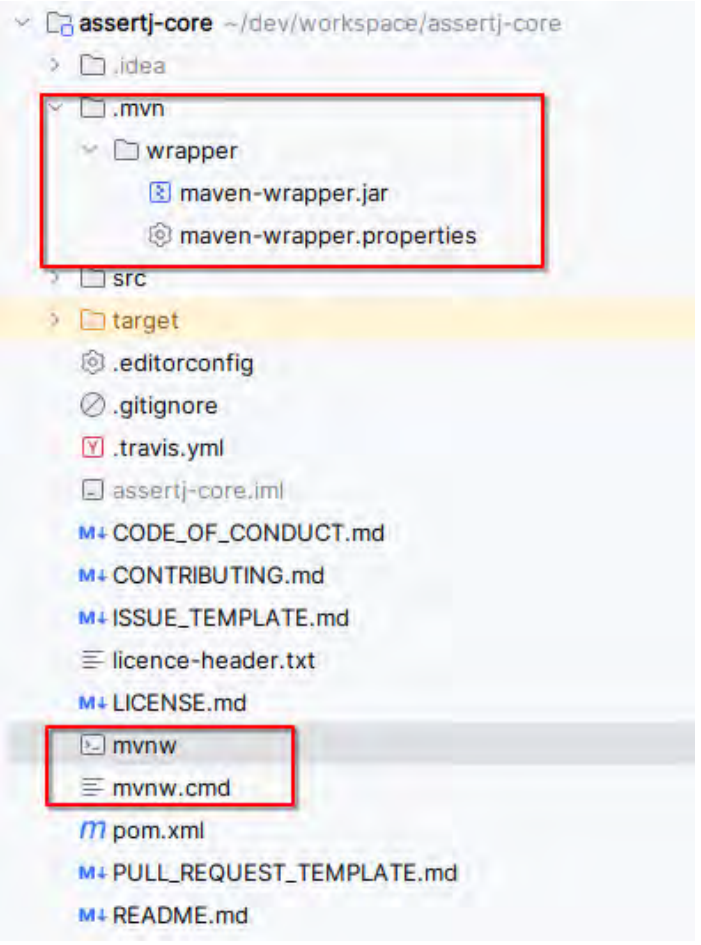

Project Structure
Project Settings
Project
Modules
Libraries
Facets
Artifacts
Platform Settings
SDKs
Global Libraries
Problems
Project
Default settings for all modules. Configure these parameters for each module on the module page as needed.
Name:
SDK:
Language level:
Compiler output:
Edit
<No SDK>
17 Eclipse Temurin 17.0.8
17 (2) Oracle OpenJDK version 17
ms-21 Eclipse Temurin 21.0.5
temurin-11 Eclipse Temurin version 11.0.17
temurin-17 Eclipse Temurin 17.0.9
temurin-17 (2) java version "17.0.9"
temurin-17 (3) Eclipse Temurin 17.0.6
temurin-21 Eclipse Temurin 21.0.5
Kotlin SDK Kotlin SDK
Download JDK...
Add JDK from disk...
Detected SDKs
~/.local/share/mise/installs/java/24.0.0 Oracle OpenJDK 24
~/.sdkman/candidates/java/24-tem Eclipse Temurin 24
~/.sdkman/candidates/java/23.0.2-tem Eclipse Temurin 23.0.2
~/.local/share/mi...ls/java/temurin-21.0.6+7.0.LTS Eclipse Temurin 21.0.6
~/.sdkman/candidates/java/21.0.6-tem Eclipse Temurin 21.0.6
~/.sdkman/candidates/java/21.0.1-tem Eclipse Temurin 21.0.1
~/.sdkman/candidates/java/11.0.19-tem Eclipse Temurin 11.0.19

Weitere Möglichkeiten

asdf

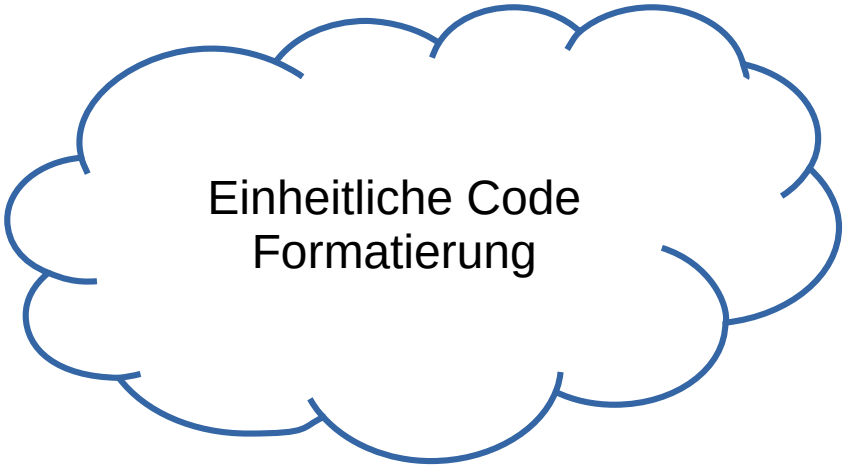
The Multiple Runtime Version Manager

Manager all your runtime versions with one tool!



Custom Installationsscripte

- Shell Scripte
- Ansible Playbooks


Einheitliche Code
Formatierung

EditorConfig

Project Page on GitHub

Follow Us on Twitter

Tweet

What is EditorConfig?

EditorConfig helps maintain consistent coding styles for multiple developers working on the same project across various editors and IDEs. The EditorConfig project consists of **a file format** for defining coding styles and a collection of **text editor plugins** that enable editors to read the file format and adhere to defined styles. EditorConfig files are easily readable and they work nicely with version control systems.

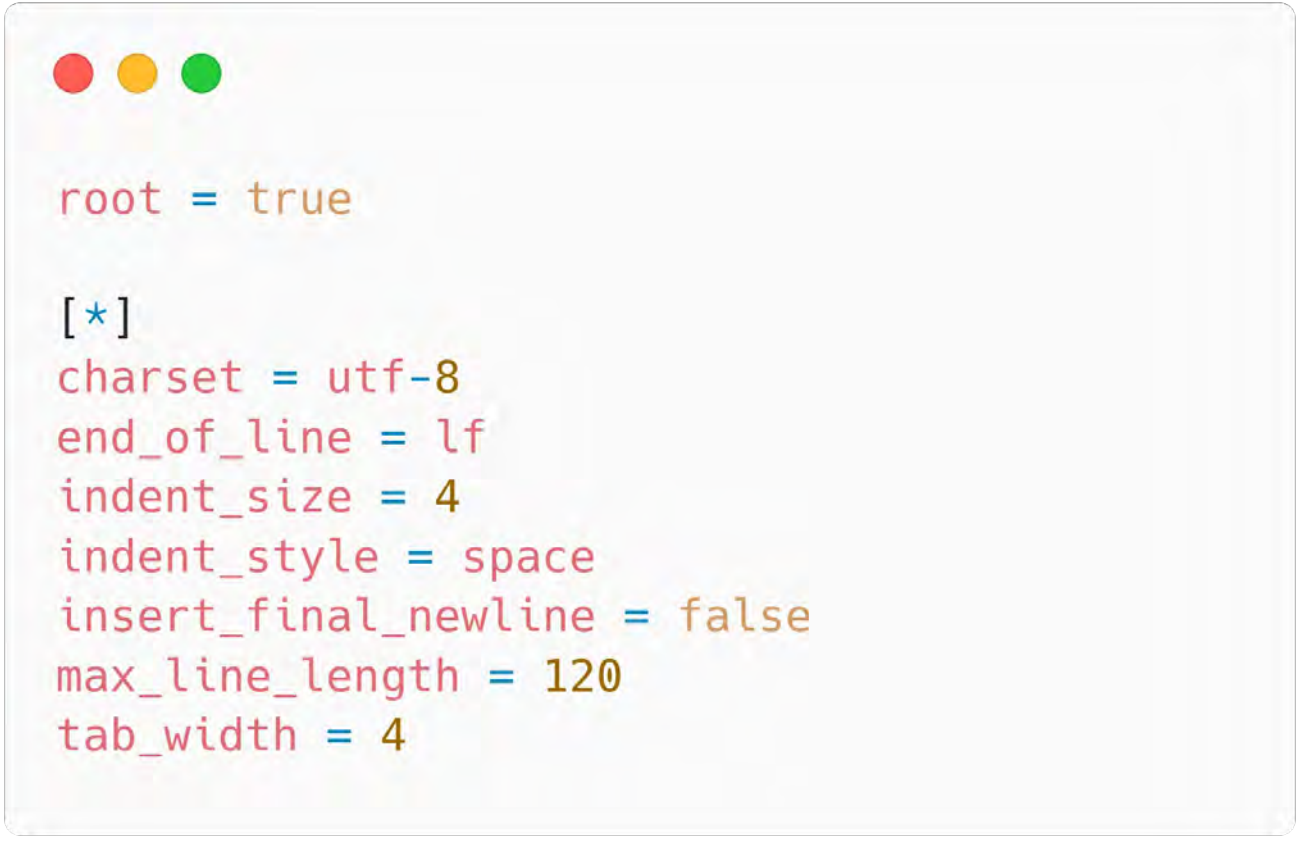
What's an EditorConfig file look like?

(A formal specification of EditorConfig is also available.)

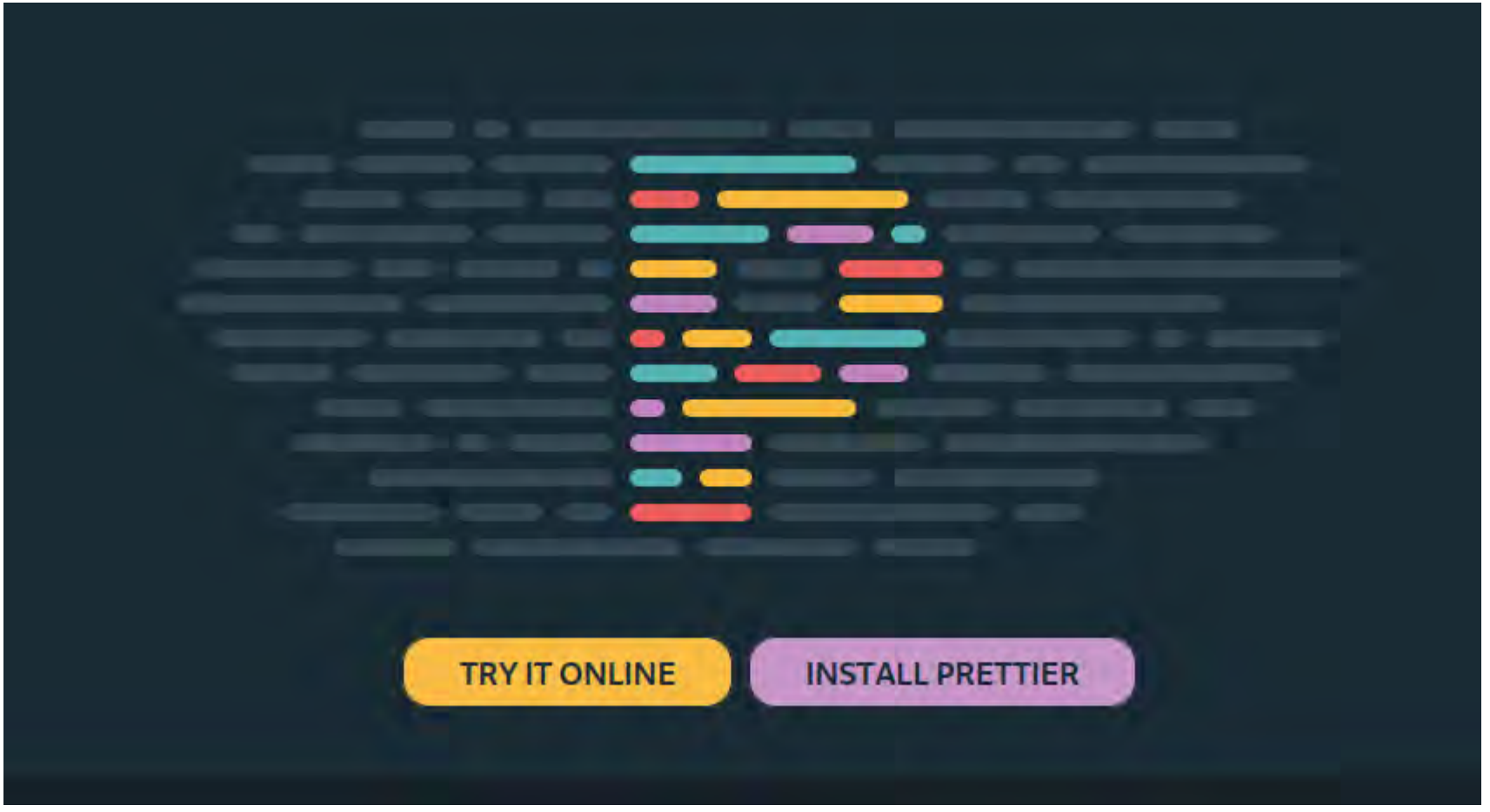
Example file

Below is an example `.editorconfig` file setting end-of-line and indentation styles for Python and JavaScript files.

```
# EditorConfig is awesome:
https://EditorConfig.org

# top-most EditorConfig file
root = true

# Unix-style newlines with a newline ending
every file
[*]
end_of_line = lf
insert_final_newline = true
```



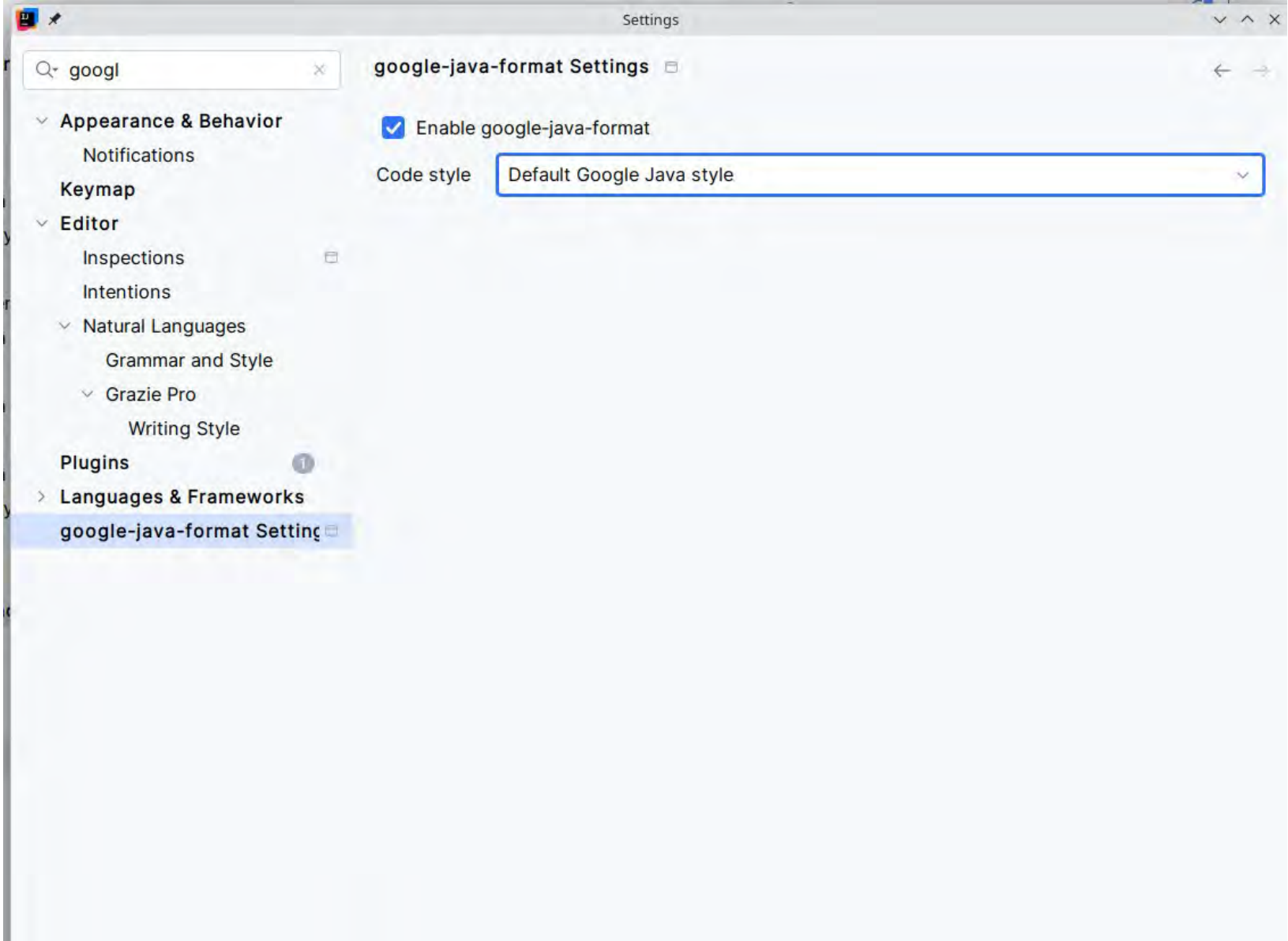
```
root = true

[*]
charset = utf-8
end_of_line = lf
indent_size = 4
indent_style = space
insert_final_newline = false
max_line_length = 120
tab_width = 4
```

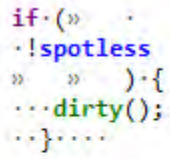

TRY IT ONLINE
INSTALL PRETTIER

Demo

google-java-format

`google-java-format` is a program that reformats Java source code to comply with Google Java Style.


Settings
googl
Appearance & Behavior
Notifications
Keymap
Editor
Inspections
Intentions
Natural Languages
Grammar and Style
Grazie Pro
Writing Style
Plugins
Languages & Frameworks
google-java-format Setting
google-java-format Settings
Enable google-java-format
Code style
Default Google Java style

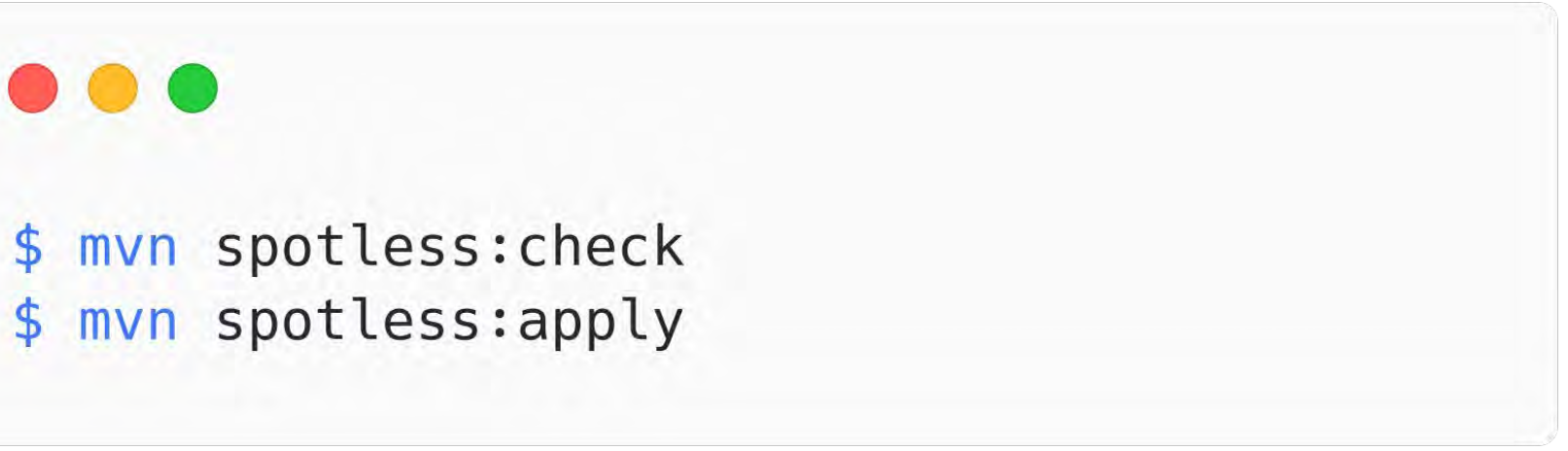


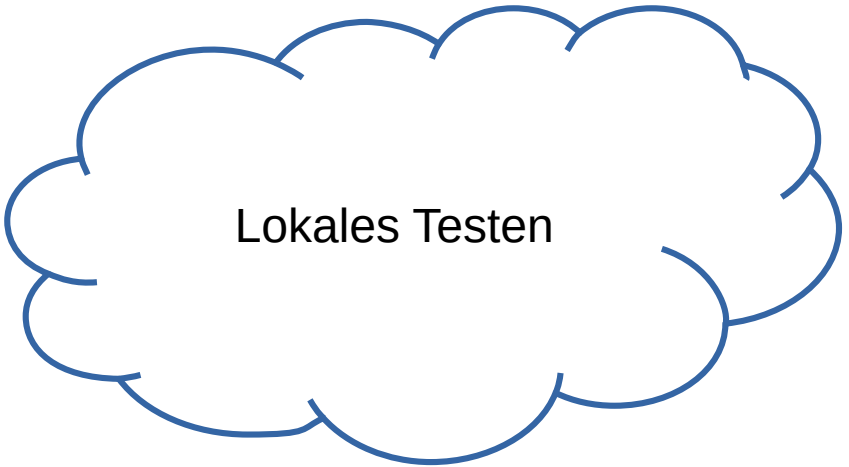
Spotless: Keep your code spotless

gradle plugin v7.0.2 | maven plugin v2.44.3 | sbt plugin 0.1.3

Spotless can format <antlr | c | c# | c++ | css | flow | graphql | groovy | html | java | javascript | json | jsx | kotlin | less | license headers | markdown | objective-c | protobuf | python | scala | scss | shell | sql | typeScript | vue | yaml | anything> using <gradle | maven | sbt | anything>.

```xml
<plugin>
    <groupId>com.diffplug.spotless</groupId>
    <artifactId>spotless-maven-plugin</artifactId>
    <version>2.44.3</version>
    <executions>
        <execution>
            <phase>process-sources</phase>
            <goals>
                <goal>check</goal>
            </goals>
        </execution>
    </executions>
    <configuration>
        <java>
            <!-- apply a specific flavor of google-java-format and reflow long strings -->
            <googleJavaFormat>
                <version>1.25.1</version>
                <style>GOOGLE</style>
                <formatJavadoc>false</formatJavadoc>
            </googleJavaFormat>
        </java>
    </configuration>
</plugin>
```


$ mvn spotless:check
$ mvn spotless:apply


Lokales Testen

Applikation lokal testen

Spring Boot Maven Plugin

Command: pnpm next dev

Demo

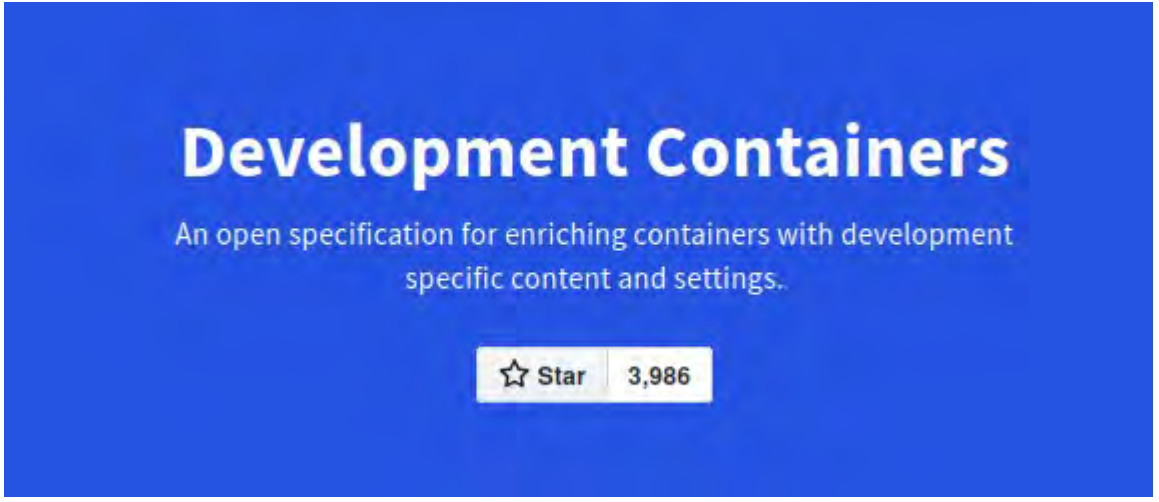
Zwischenfazit

- ✔ Native Lösungen
- ✔ Viele Möglichkeiten
- ✔ Niedrige Ramp Up Zeit
- ✔ Zum Teil gute Integration in den IDEs

- ✗ Native Lösung hauptsächlich für Linux und Mac
- ✗ Windows über WSL
- ✗ Keine One-fits-all Lösung
- ✗ Hoher Initial-Setup

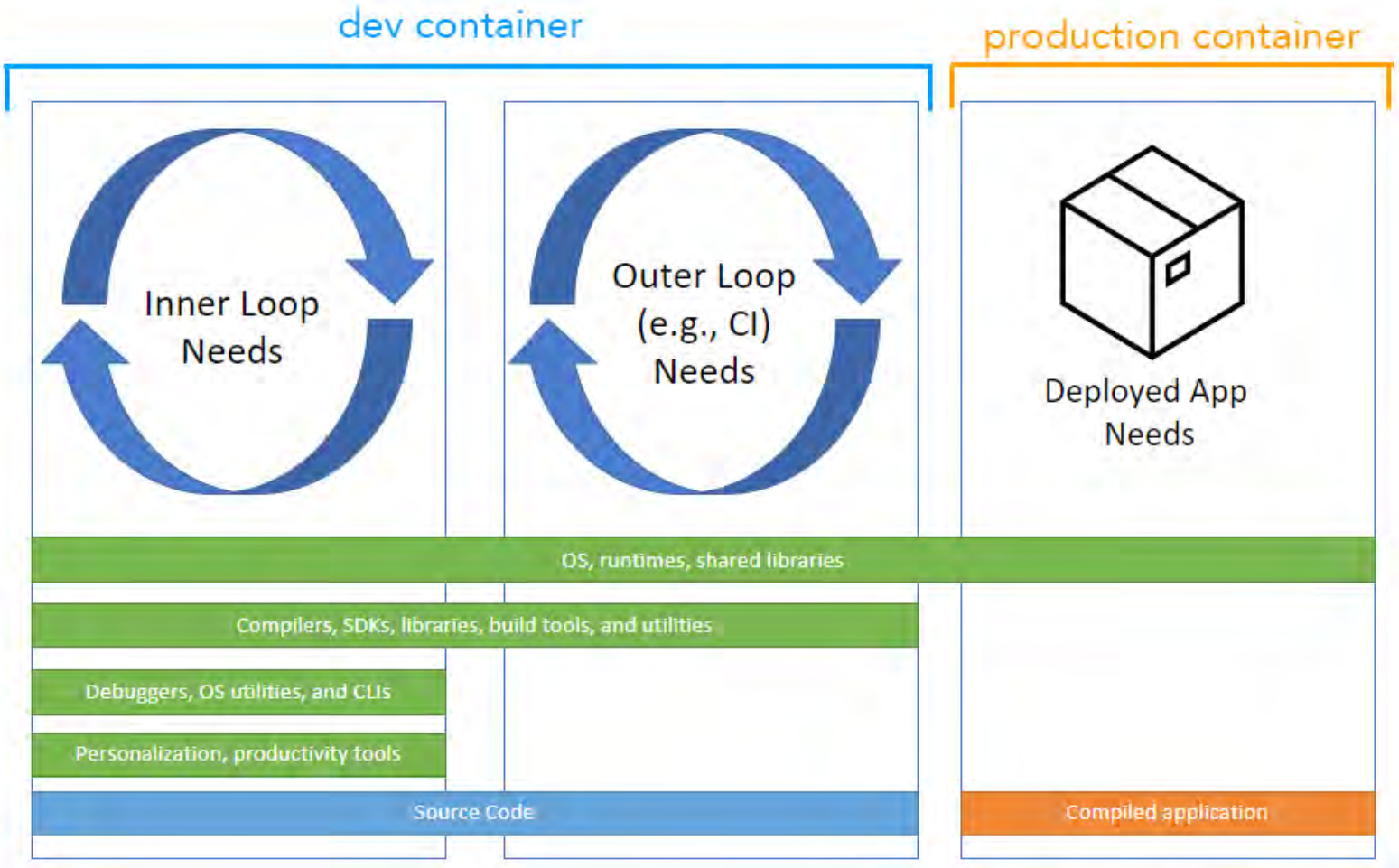
Lösungansatz: Container-basiert

Dev Containers



https://containers.dev/

Idee


dev container
production container
Inner Loop
Needs
Outer Loop
(e.g., CI)
Needs
Deployed App
Needs
OS, runtimes, shared libraries
Compilers, SDKs, libraries, build tools, and utilities
Debuggers, OS utilities, and CLIs
Personalization, productivity tools
Source Code
Compiled application

Demo

Zwischenfazit

- ✔ One-fits-all-Lösung
- ✔ Technologie-Stack unabh.
- ✔ 100% Container basiert

- ✗ Wenige IDE unterstützen es
- ✗ Ramp up Zeit hoch
- ✗ Initialaufwand
- ✗ (Microsoft)

Lösungsansatz: Reproduzierbare Umgebung ohne Container

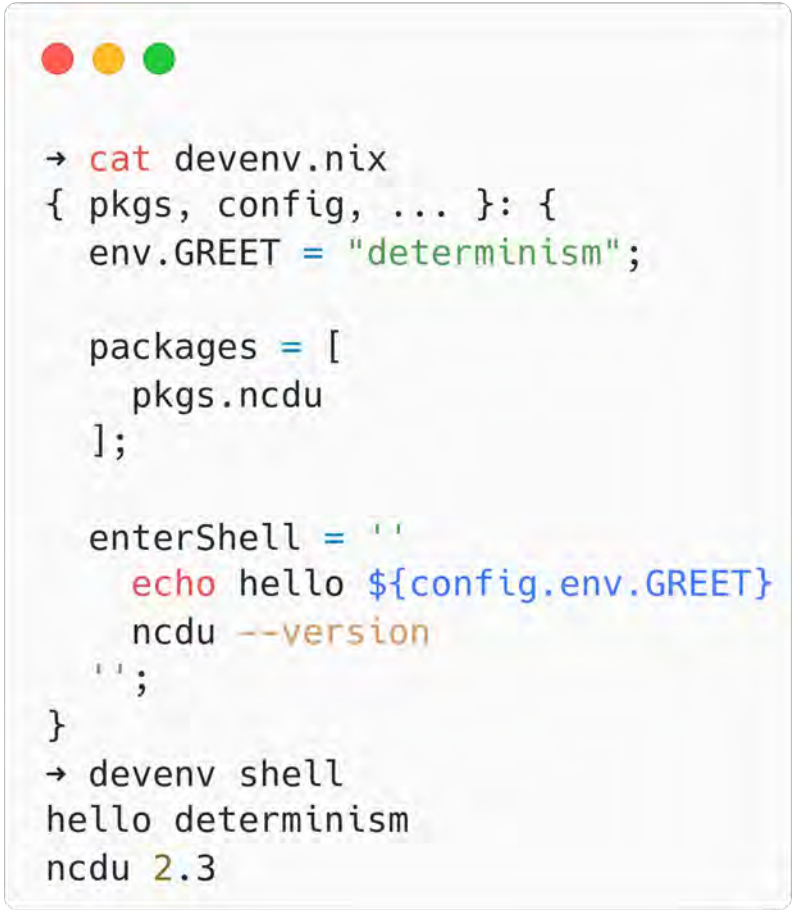
Basis - NixOS

- Linux-Distro, die deklarativ konfiguriert wird
- Hauptprinzipien:
  - Deklarative Konfiguration
  - Atomare Updates und Rollbacks
  - Paketisolierung
  - Reproduzierbarkeit

Paketmanager Nix

- Jedes Paket, das installiert wird bekommt eine eindeutige Kennung
- Kennung abh. Version und Konfiguration
- Vereinfacht parallele Installation von gleiche Paket in verschiedenen Versionen


devenv



```
→ cat devenv.nix
{ pkgs, config, ... }: {
  env.GREET = "determinism";

  packages = [
    pkgs.ncdu
  ];

  enterShell = ''
    echo hello ${config.env.GREET}
    ncdu --version
  '';
}
→ devenv shell
hello determinism
ncdu 2.3
```

Jetify Devbox

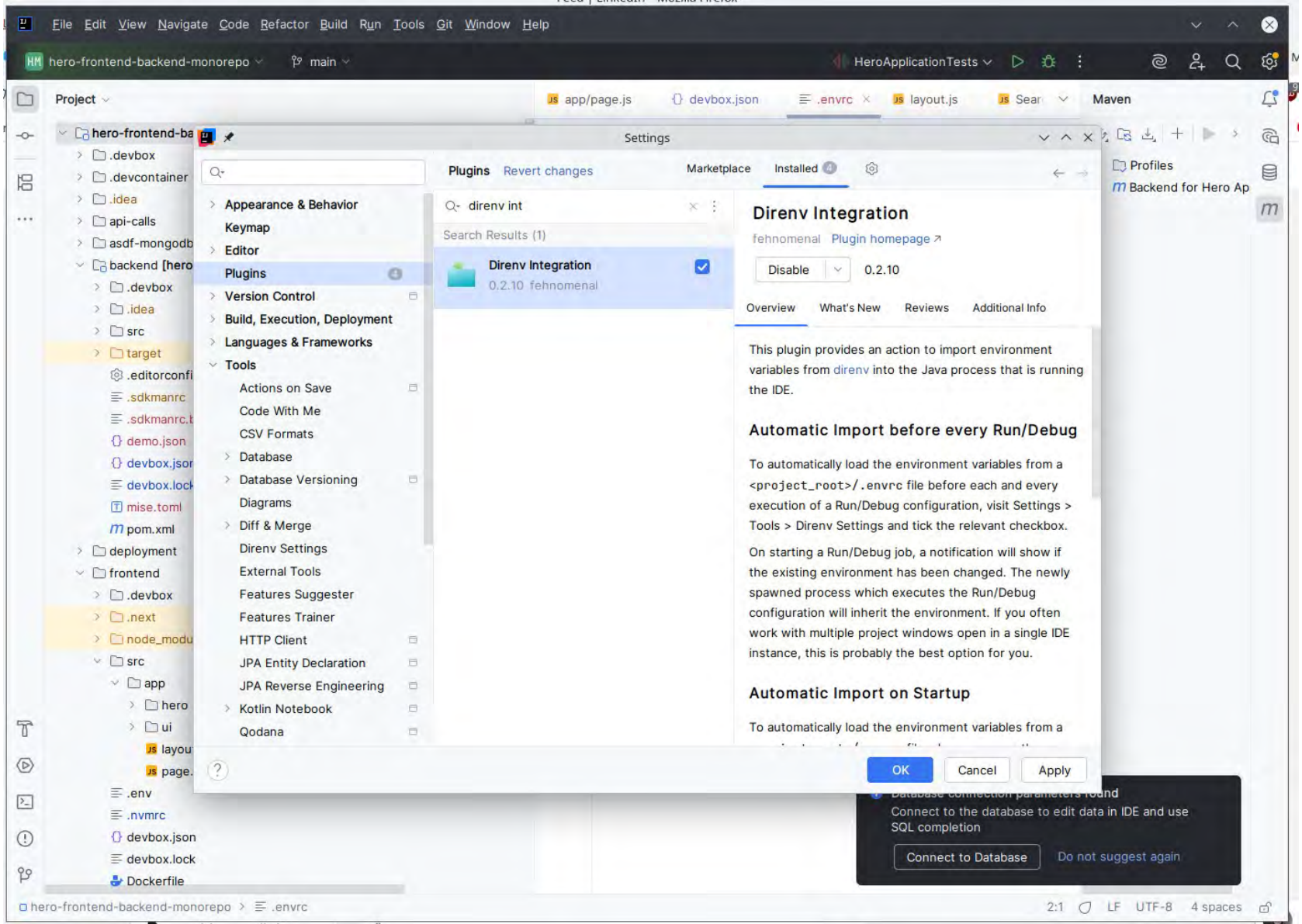
Demo

IDE Integration via direnv

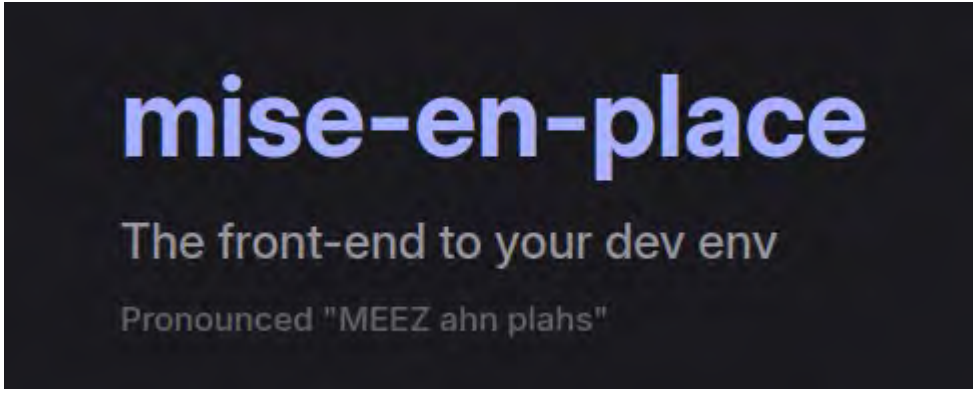
direnv

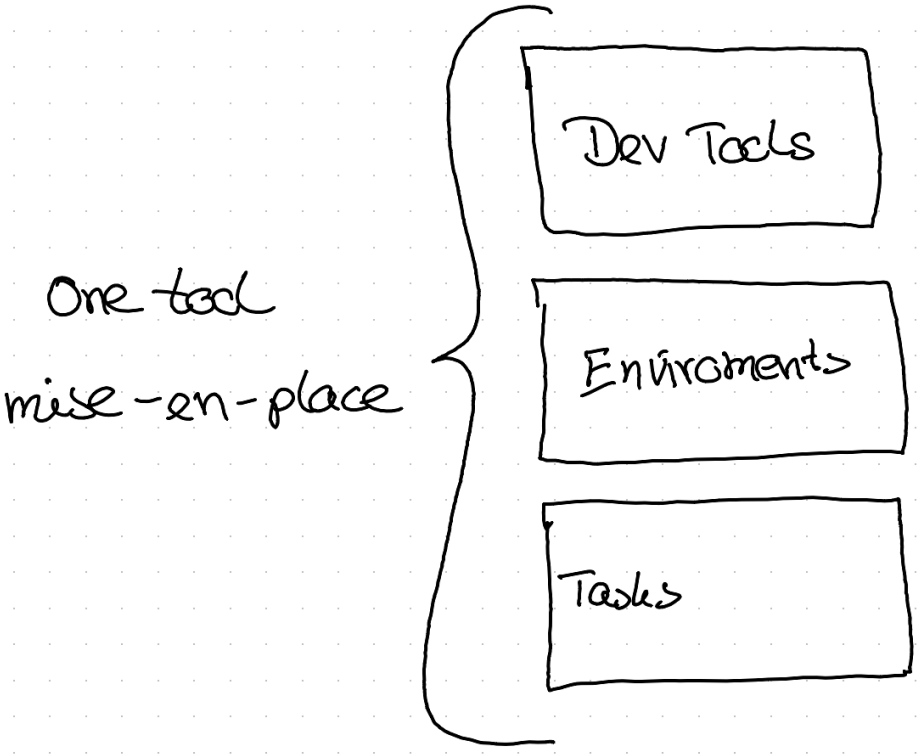
direnv – unclutter your .profile

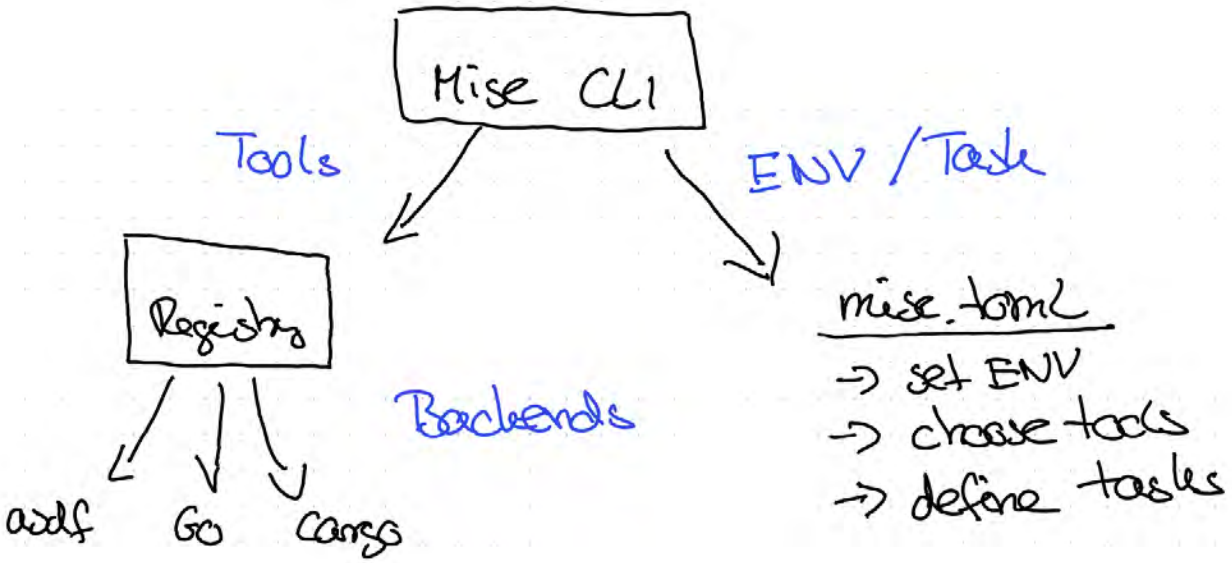
Demo


File Edit View Navigate Code Refactor Build Run Tools Git Window Help
hero-frontend-backend-monorepo
main
HeroApplicationTests
Project
app/page.js
devbox.json
.envrc
layout.js
Maven
Settings
Plugins
Revert changes
Marketplace
Installed
direnv int
Search Results (1)
Direnv Integration
0.2.10 fehnomenal
Direnv Integration
fehnomenal
Plugin homepage
Disable
0.2.10
Overview
What's New
Reviews
Additional Info
This plugin provides an action to import environment variables from direnv into the Java process that is running the IDE.
Automatic Import before every Run/Debug
To automatically load the environment variables from a <project_root>/.envrc file before each and every execution of a Run/Debug configuration, visit Settings > Tools > Direnv Settings and tick the relevant checkbox.
On starting a Run/Debug job, a notification will show if the existing environment has been changed. The newly spawned process which executes the Run/Debug configuration will inherit the environment. If you often work with multiple project windows open in a single IDE instance, this is probably the best option for you.
Automatic Import on Startup
To automatically load the environment variables from a
Appearance & Behavior
Keymap
Editor
Plugins
Version Control
Build, Execution, Deployment
Languages & Frameworks
Tools
Actions on Save
Code With Me
CSV Formats
Database
Database Versioning
Diagrams
Diff & Merge
Direnv Settings
External Tools
Features Suggester
Features Trainer
HTTP Client
JPA Entity Declaration
JPA Reverse Engineering
Kotlin Notebook
Qodana
OK
Cancel
Apply
Connect to the database to edit data in IDE and use SQL completion
Connect to Database
Do not suggest again
Profiles
Backend for Hero Ap
hero-frontend-backend-monorepo > .envrc
2:1
LF
UTF-8
4 spaces

Alternative ohne Nix



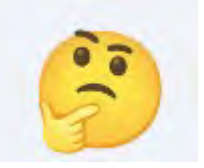

One tool
mise-en-place
Dev Tools
Enviroments
Tasks


Mise CLI
Tools
ENV / Task
Registry
Backends
asdf
Go
cargo
mise.toml
-> set ENV
-> choose tools
-> define tasks

Demo

Zwischenfazit

- ✔ One fits all Lösung auf nativer Basis
- ✔ Kein Container Overhead
- ✗ Integration nur über 3rd Party Libs
- ✗ Kein nativen Support für Windows

Und was nutzt jetzt Sandra? 

Was ist mit Cloud-basierten Lösungen?

Fragen?

mail@sandra-parsick.de
@sparsick@mastodon.social

https://codeberg.org/sparsick/dev-env-talk