

Exploring WebAssembly for versatile plugin systems
through the example of a text editor

Florian Hartung

10.03.2025

Plugin Systeme sind überall

- IDEs (z.B. ein Quellcode-Formatter, Preview für Webseite)
- Audio (z.B. VSTs bei Musikproduktion)
- Computerspiele (z.B. Minecraft Server)
- Browsererweiterungen (z.B. Adblocker, Dark Mode)
- ...

Probleme mit Text Editoren

- Plugins oft nicht abgesichert
 - ▶ Beispiel: Issue VSCode offen seit 2018¹. Scheitert an Sandboxed execution
- Nicht ressourcen-effizient (JS, Java, Lua)
- Plugin-Entwicklung erfordert Einarbeitung in eine neue Sprache

¹<https://github.com/microsoft/vscode/issues/52116>

WebAssembly

- Web + Assembly (quasi: Assembly für das Web bzw. den Browser eines Clients)
- Compilation target (Abbildung)
- schnell & sicher
- Ermöglicht HPC im Browser, wo JS nicht mehr ausreicht ohne Risiko für den Client

WebAssembly für Plugin Systeme

1. Technologievergleich
2. Einordnen von WebAssembly

	VSCode	IntelliJ	Zed	ZelliJ	Wasm	Wasm (WIT)
Performance						
Plugin size						
Isolation/Safety	2	0	5	5	5	5
Portability	3	3	4	4	4	4
Erweiterbarkeit	2	2	5	2	1	5

Proof of concept

- Implementierung eines einfachen Plugin Systems für den Helix Editor²

²<https://www.helix-editor.com>

Bla