

# GoWASM の紹介

2025/06/01

WebページやJavaScript LT会

Tam@XM研究員

# 自己紹介

福山雅治

自己紹介の長い奴は何を語らせても駄目

# アンケート

動きのある Web ページを作るために一番使っている（もしくは使えそうな）言語はどれですか？

1. Flash
2. SVG
3. JavaScript

## 結論

その JavaScript がオワコンという話を展開します。

そもそも this オブジェクトの挙動を正しく説明できますか？

(私は出来ません。)

## 問題点

JavaScript は型なし言語と主張しています。

123 と "123" の区別が曖昧になってるところが多く、

12 + 3

が 15 になったり 123 になったり。

→ 盛大なバグの温床になっています。

## 誤解

そもそも JavaScript という名前自体、混同の元になっています。

**Java と JavaScript は はなまるうどん と 丸亀製麺 くらい違う。**

# AltJS

JavaScript の置き換えを目的とした言語がたくさん出来ました。

その急先鋒にあるのが TypeScript です。

TypeScript は JavaScript に変数型定義を追加していますが、JavaScript の文法を損なうことなく自然に拡張できています。

また、TypeScript からコンパイルすることによって、JavaScript に変換できます。

そのため、JavaScript が動く環境であれば、どこでも TypeScript は動かせます。

## TypeScript の問題点

JavaScript の文法を拡張しているため、 JavaScript の仕様に引きずられてしまいます。

JavaScript を拡張しているので、文法が肥大化しており、覚えることがたくさんあります。



# 代替案

WebAssembly（通称 Wasm）という統一規格が出来ています。

Gemini によると：

- ウェブブラウザ上で高速に動作するバイナリコードの仕様
- JavaScriptのパフォーマンスを補完
- ネイティブに近いパフォーマンスで動作
- セキュリティを重視して設計されている
- ゲーム開発や動画処理など、高速処理が求められるWebブラウザアプリケーションに最適

## どういうコード？

Web **Assembly** という名前だけあって、中身はマシン語（つまりバイナリ）です。

通常は C言語や Rust からコンパイルしてバイナリを作成します。

作るの余計大変では？

# 解決策

Go言語から Wasm バイナリにコンパイルすることが出来ます。

Go言語は

- 必要最低限の命令しか定義されていないので、覚えることが少ない
- 並列処理も言語レベルでサポート
- 様々なツールやライブラリが揃っている

ので作るのが簡単です。

## デメリット

- WebAssembly が定義しているのは基本演算命令のみ
- ネットワークアクセスや画面書き換えなどは出来ない
- 外部とのやり取りは JavaScript などを経由して行わないといけない

# デモ