WebAssembly - The Next Big Platform

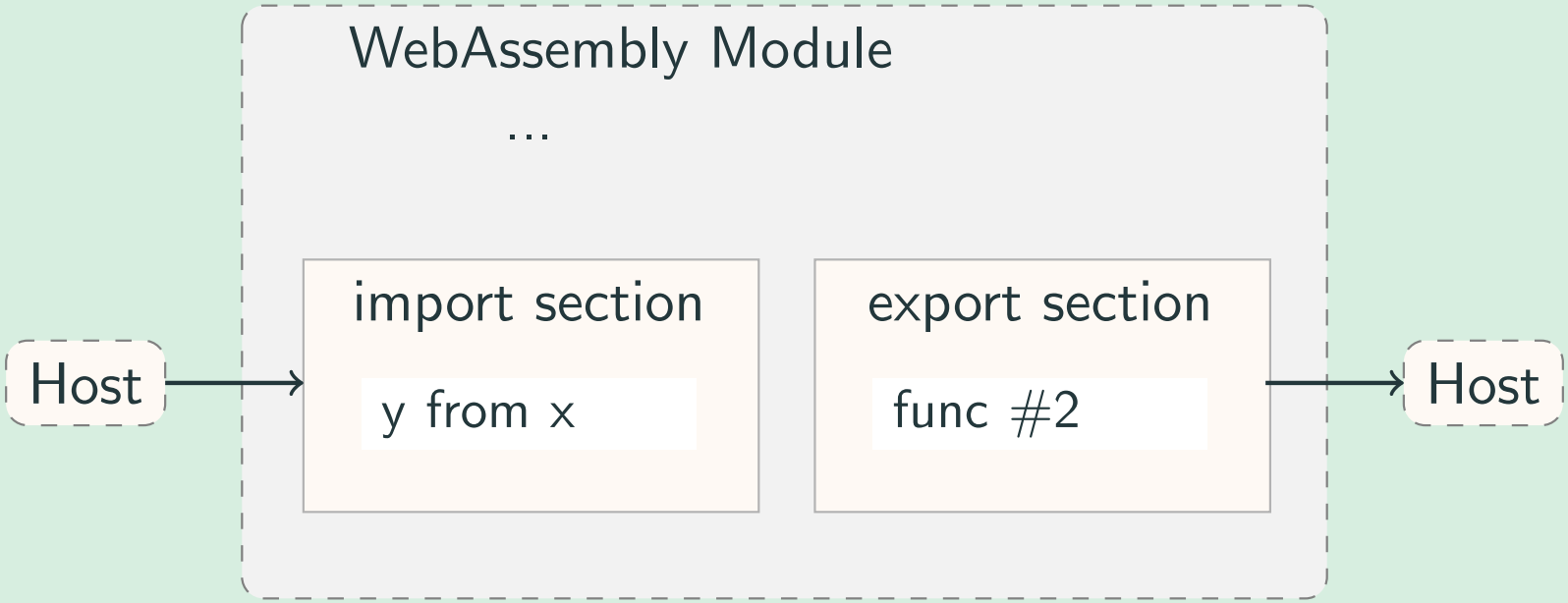
Sven Sauleau

2019

Sven Sauleau

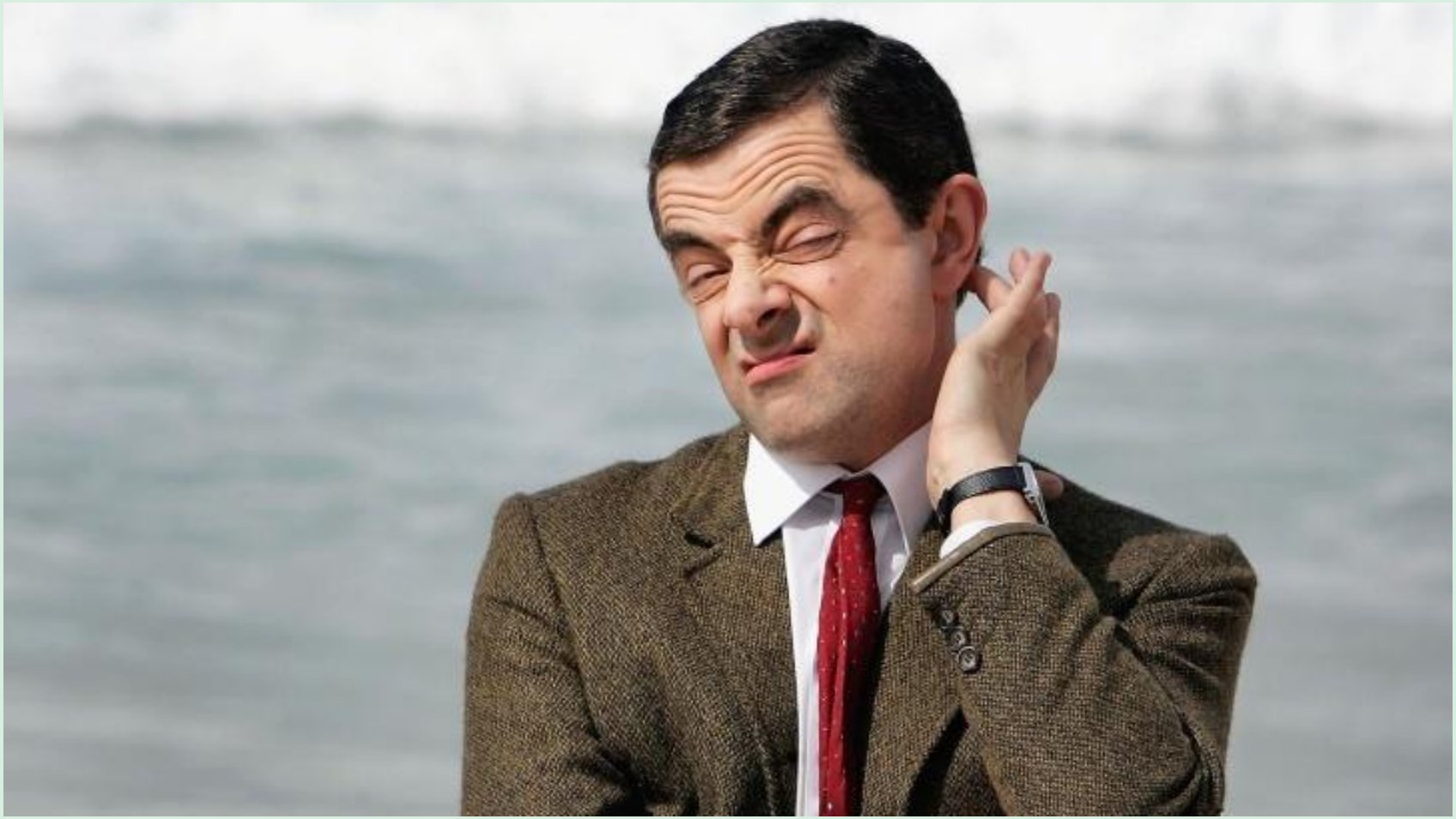
API

```
1  const imports = {
2      env: {
3          one: 1
4      }
5  };
6
7  const {instance} = await WebAssembly.instantiate(
8      module, imports);
9
10 intance.exports.someExport();
```


WebAssembly Module
...
import section
y from x
export section
func #2
Host
Host

4 basic value types, all of which are available in common hardware.

$$\begin{aligned} \text{value type} &::= \text{i32} \mid \text{i64} \mid \text{f32} \mid \text{f64} \\ \text{function type} &::= [\mathit{vec}(\text{value type})] \to [\mathit{vec}(\text{value type})] \end{aligned}$$



```
const imports = {
    env: {
        setElementHeight(value) {
            document
                .getElementById("foo")
                .style.height = value;
        }
    }
};

// ...
```

Rust and WebAssembly

rustwasm/wasm-bindgen

“Facilitating high-level interactions between wasm modules and JavaScript”

```
let element: Element = // ...
element.set_inner_html("hi");
```

```
pub fn some_export() -> MyStruct {
    MyStruct {}
}
```


Wasm
JavaScript

rustwasm/gloo

"A modular toolkit for building fast, reliable Web applications and libraries with Rust and Wasm"

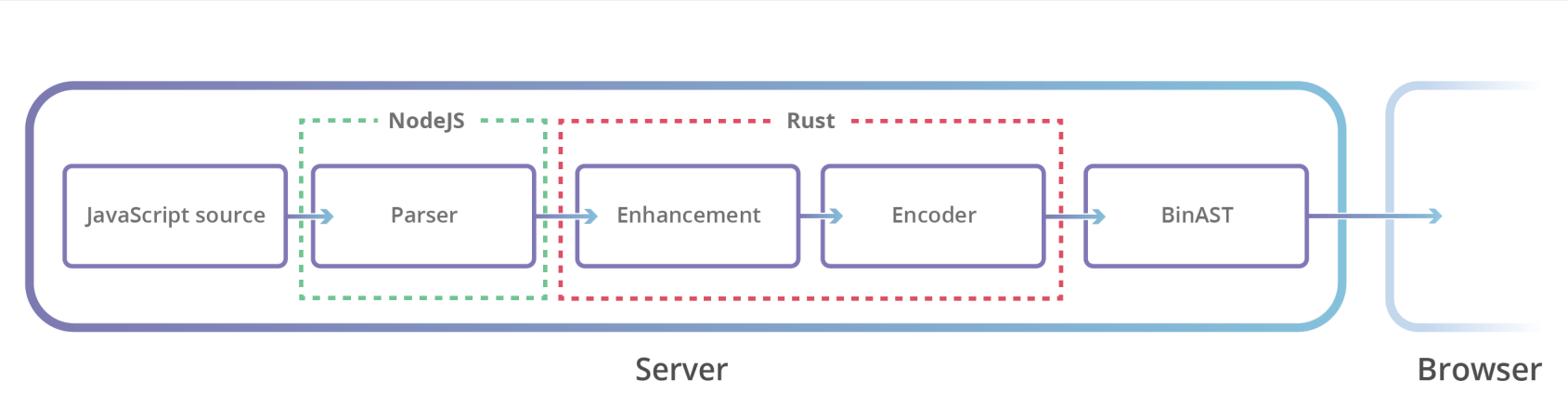
gloo-timers [1]

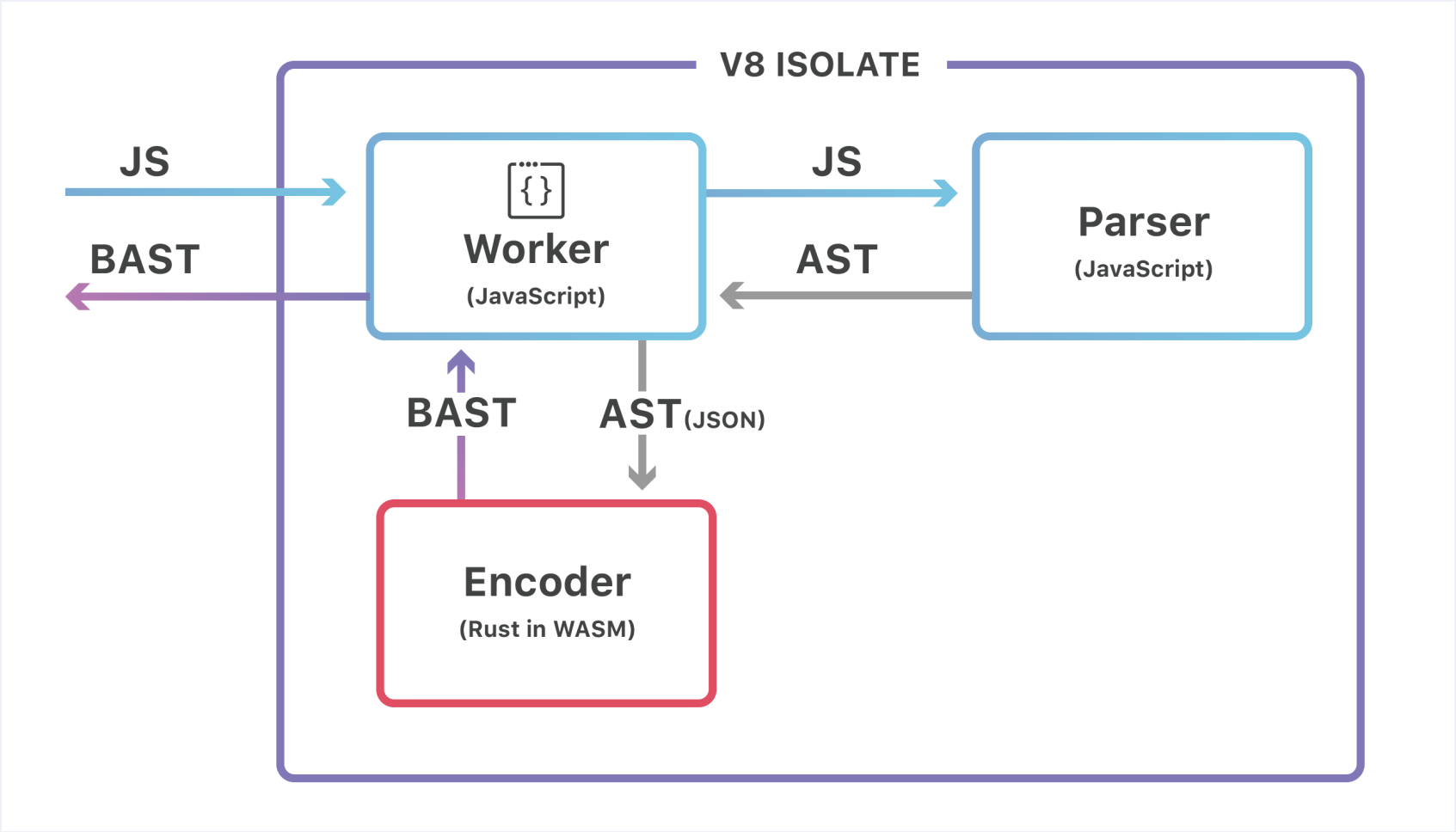
```
use gloo_timers::callback::Timeout;

let timeout =
    Timeout::new(1_000, move || {
        // Do something after the one
        // second timeout is up!
    });
```

[1] Working with timers on the Web: setTimeout and setInterval

Cloudflare BinAST

"Compression", binary format for JavaScript.


NodeJS
Rust
JavaScript source
Parser
Enhancement
Encoder
BinAST
Server
Browser


V8 ISOLATE
JS
BAST
Worker
(JavaScript)
JS
AST
Parser
(JavaScript)
BAST
AST (JSON)
Encoder
(Rust in WASM)

demo!

Thanks!