JAVA DE REGRESO A LA WEB

CON A WEBASSEMBLY

MIGUEL USECHE @ JCONF DOMINICANA | JUNIO | 2019

¡HOLA! SOY MIGUEL

- Desarrollador web independiente y profesor universitario.
- Voluntario en comunidades de código abierto:
- Mozillero de 🇻🇪 viviendo en 🇨🇴

San Cristobal

Táchira
Venezuela

Directions

SAVE | NEARBY | SEND TO YOUR PHONE | SHARE

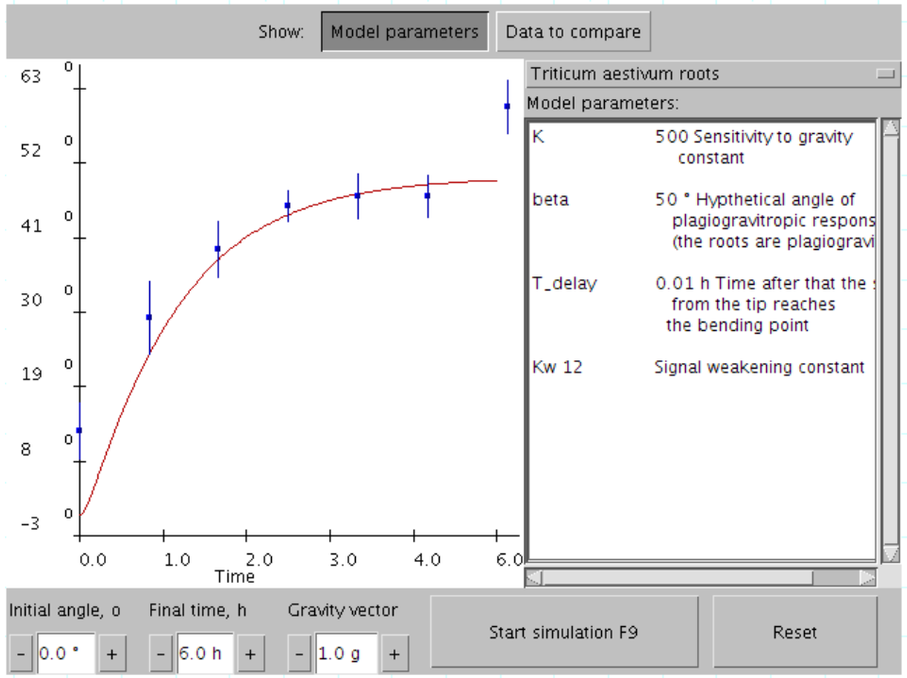
Quick facts

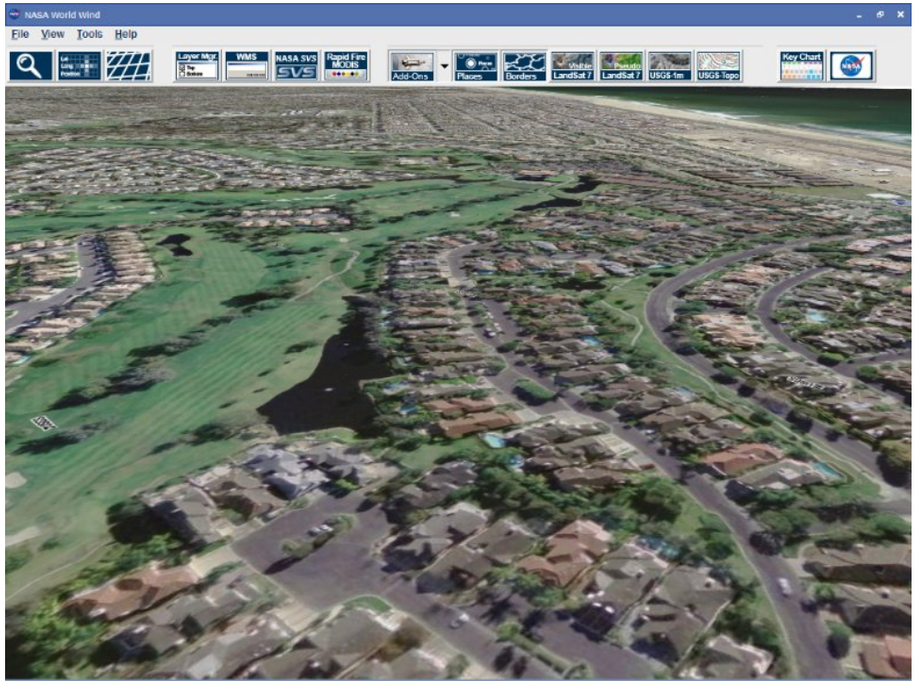
San Cristóbal is the capital city of the Venezuelan state of Táchira. It is located in a mountainous region of Western Venezuela. The city is situated 818 metres above sea level in the northern Andes overlooking the Torbes River, 56 kilometres from the Colombian border. Wikipedia

¡EMPECEMOS!


BACK to the 90s
SANDY & PAPO
GAME BOY COLOR

JAVA APPLETS





Aplicación matemática

NASA World Wind

¿POR QUÉ DESAPARECIERON LOS JAVA APPLETS?

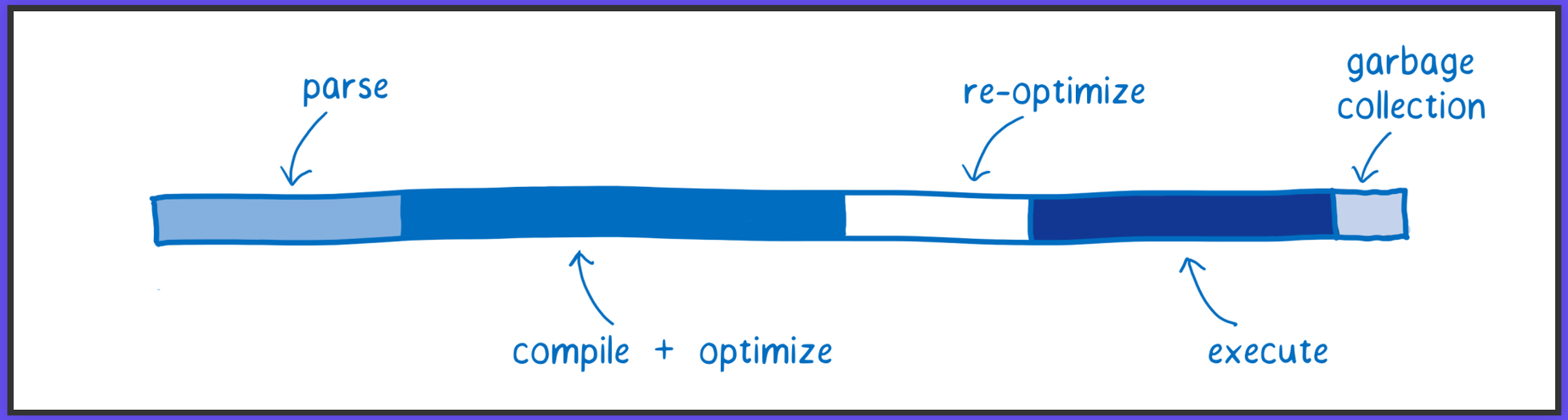
- Requeria instalar Java en tu equipo.
- Nula integración con la web.
- Perdió fuerza debido a Flash, Silverlight, HTML5
- Problemas en dispositivos móviles
- ¿Problemas de seguridad?

....LOS APPLETS DE JAVA FUERON DESPLAZADOS...

....ante JavaScript


RENDIMIENTO

CÓMO LOS NAVEGADORES EJECUTAN JS



Fuente: Lin Clark © 2017.

VEAMOS UN EJEMPLO...

UNA SUMA EN JS

```
function suma(a, b) {
    return a + b;
}
```

Sencillo, ¿no?

Debería ***"ser una única"*** instrucción del CPU

CÓMO ECMA-262 DEFINE UNA ADICIÓN

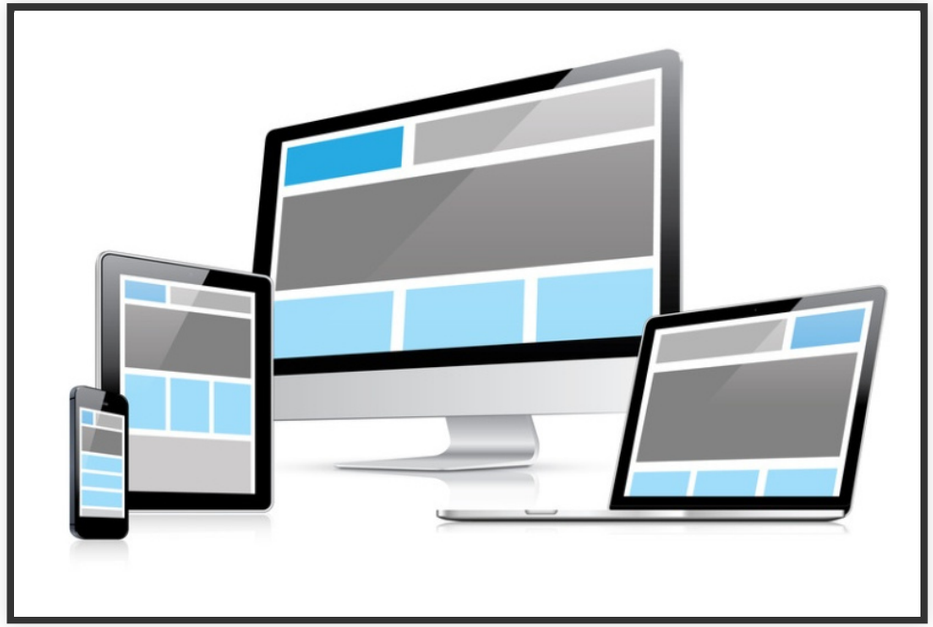
1. *Let lref be the result of evaluating AdditiveExpression.*
2. *Let lval be GetValue(lref).*
3. *Let rref be the result of evaluating MultiplicativeExpression.*
4. *Let rval be GetValue(rref).*
5. *Let lprim be ToPrimitive(lval).*
6. *Let rprim be ToPrimitive(rval).*
7. *If Type(lprim) is String or Type(rprim) is String, then*
    - *Return the String that is the result of concatenating ToString(lprim) followed by ToString(rprim)*
    - *Return the result of applying the addition operation to ToNumber(lprim) and ToNumber(rprim).*

Fuente: ECMAScript® Language Specification. (Sec. 11.6.1)

¿POR QUÉ?

Hay que hacer sacrificios

Entienden lenguajes de programación

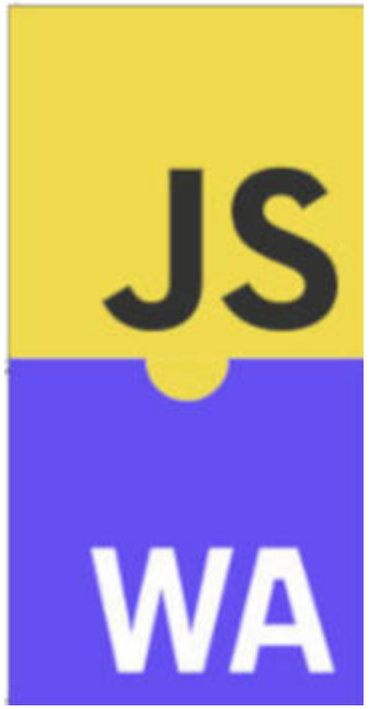


Entienden binario o código de bajo nivel

¿CÓMO RESOLVER ESTO?

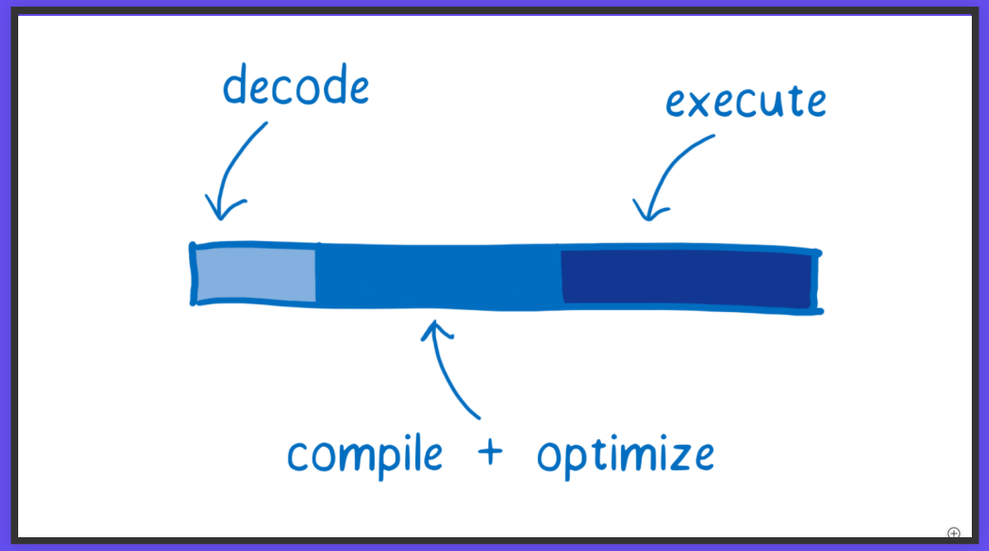
WEBASSEMBLY

WEBASSEMBLY



- Formato binario
- **NO** remplaza JS
- Interface desde/hacia tu lenguaje (Java)
- Integración con WebAPI
- float32,int64, threads, SIMD
- Fácil de compilar, verificar y extensible
- Bloques de memoria de 1 byte

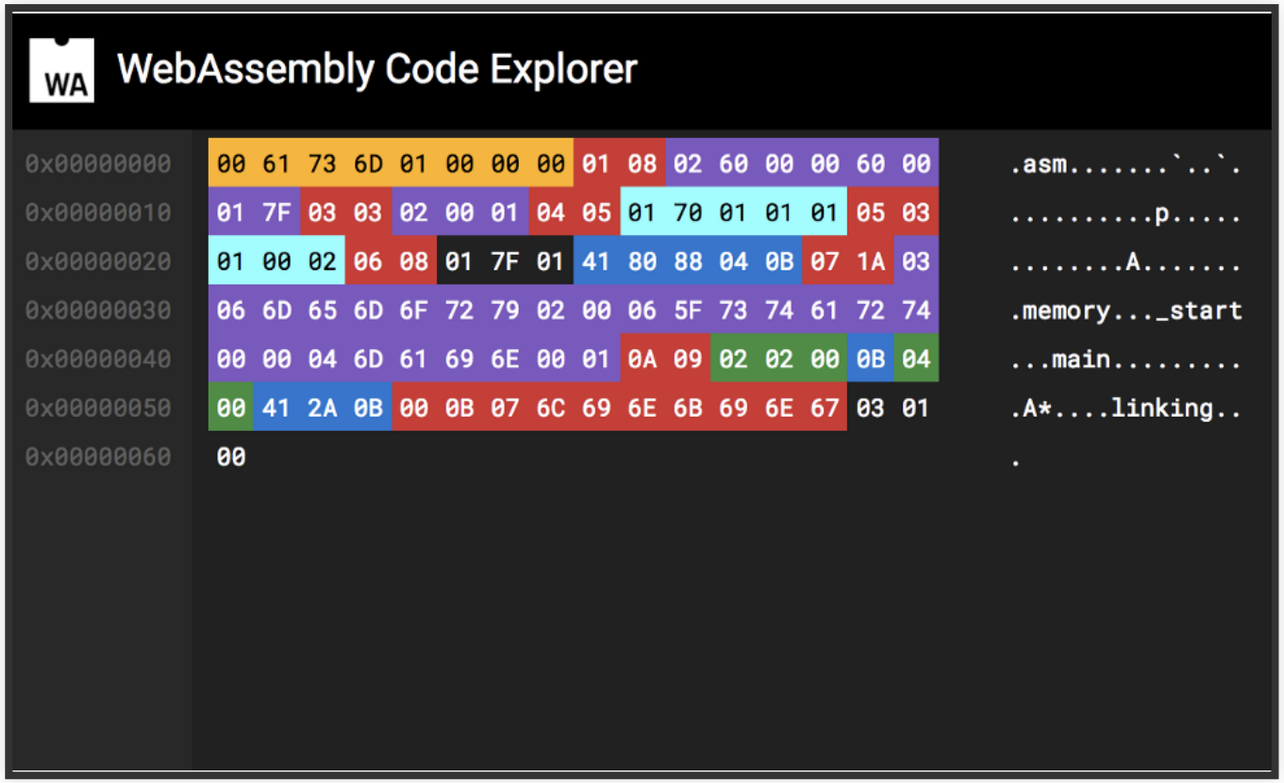
CÓMO LOS NAVEGADORES EJECUTAN WASM



Fuente: Lin Clark © 2017.

FORMATOS DE WASM

FORMATO BINARIO



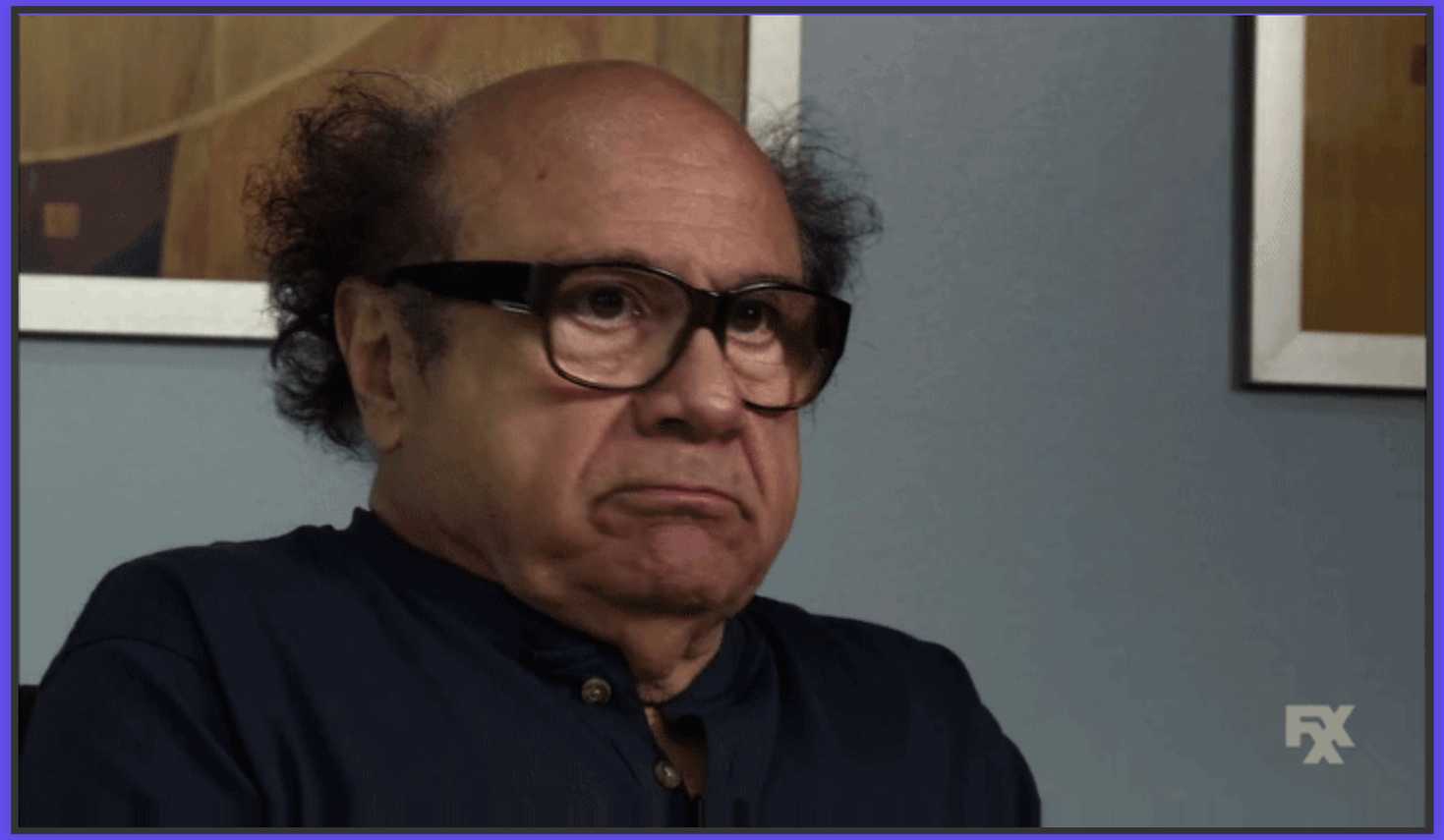
Puedes verlo con WASM Code Explorer

WEBASSEMBLY TEXT FORMAT

```
(module
  (type $type0 (func (param i32)))
  (type $type1 (func))
  (import "sys" "print" (func $import0 (param i32)))
  (memory (;0;) 200 200)
  (export "memory" (memory 0))
  (export "main" (func $func1))
  (func $func1
    i32.const 0
    call $import0
  )
  (data (i32.const 0)
    "Hello, world\00"
  )
)
```

Hola Mundo en WAT

TAL VEZ PIENSES QUE ES UNA MODA PASAJERA....



¡PERO NO!
¡TIENE GRAN SOPORTE!

¿CÓMO SE CREA?

WEBASSEMBLY EN JAVA

Existen 3 proyectos:

- JWebAssembly
- TeaVM
- Bytecoder

WEBASSEMBLY EN JAVA

COMPILACIÓN

- JWebAssembly y TeaVM

```
Generados con MAVEN
```

- Bytecoder:

```
java -jar bytecoder-cli-2019-06-13-executable.jar
 -classpath=. -mainclass=TU_PAQUETE.ArchivoPrincipa
 -builddirectory=. -backend=js -minify=false
```

¿KOTLIN?

KOTLIN A WASM CON EMSCRIPTEN

Emscripten es conjunto de herramientas para llevar lenguajes de alto nivel a WASM

Kotlinsup → LLVM → Emscripten + Binaryen → WASM

* Todo este proceso se hace automáticamente con un script de Graddle.

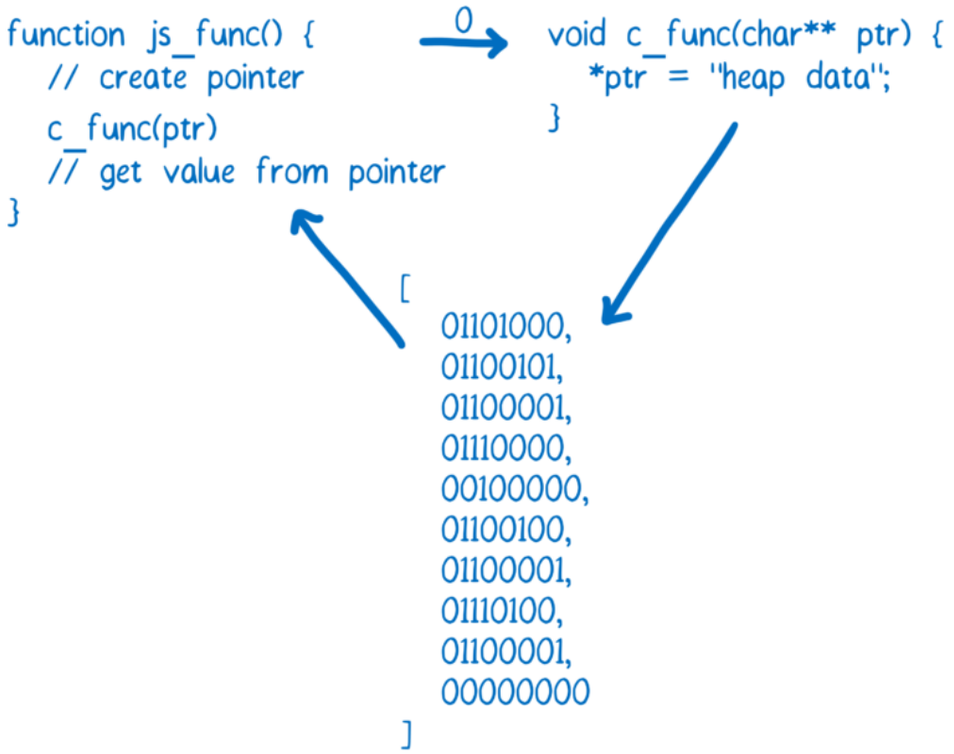
¿QUÉ OBTENGO CON ESTO?

RESULTADOS DE LA COMPILACIÓN

Vas a obtener 3 archivos:

1. El archivo **.wasm**.
2. El código de JS para importar el módulo.
3. Una página HTML para ejecutar el módulo.

INTERACCIÓN DEL CÓDIGO



Comunica código entre JS y WASM

INTERACCIÓN JAVA -> JS

JWEBASSEMBLY

Java

```java
import de.inetsoftware.jwebassembly.api.annotation.Export;

@Export
public static int sumaEnJava( int a, int b ) {
    return a + b;
}
```

JavaScript

```javascript
let funcionJs;

WebAssembly.instantiateStreaming(fetch('simple.wasm'))
.then(obj => funcionJs = obj.instance.exports.sumaEnJava());
```

INTERACCIÓN JAVA → JS

BYTECODER

Java

```java
public class MiClase {
    @Export("funcionDeJava")
    public static void funcionDeJava() {
    }
}
```

JavaScript

```javascript
bytecoder.exports.funcionDeJava();
```

INTERACCIÓN JS → JAVA

JWEBASSEMBLY

Re-implementación:

```
import de.inetsoftware.jwebassembly.api.annotation.Import;

@Import( module = "global.Math", name = "max" )
static int max( int a, int b) {
    return Math.max( a, b );
}
```

Usando native:

```
import de.inetsoftware.jwebassembly.api.annotation.Import;

@Import( module = "global.Math", name = "max" )
public static native double max( int a, int b);
```

INTERACCIÓN JS → JAVA

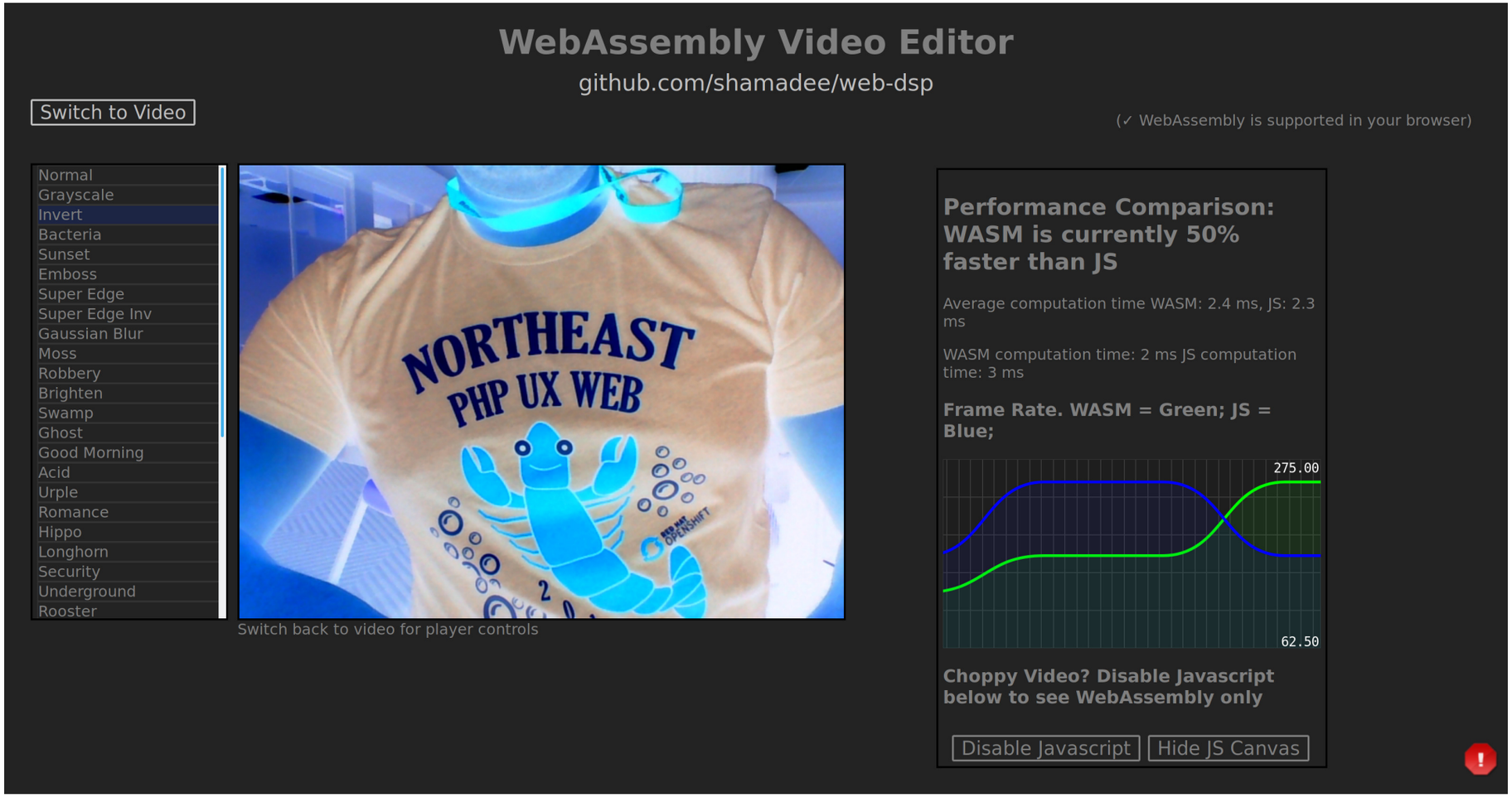
BYTECODER

```
import de.mirkosertic.bytecoder.api.Import;

public class CanvasRenderingContext2D {

  @Import(module = "math", name = "max")
  public static native long max(long aValue1, long aValue2);

}
```

¿QUÉ PUEDO HACER CON WASM?

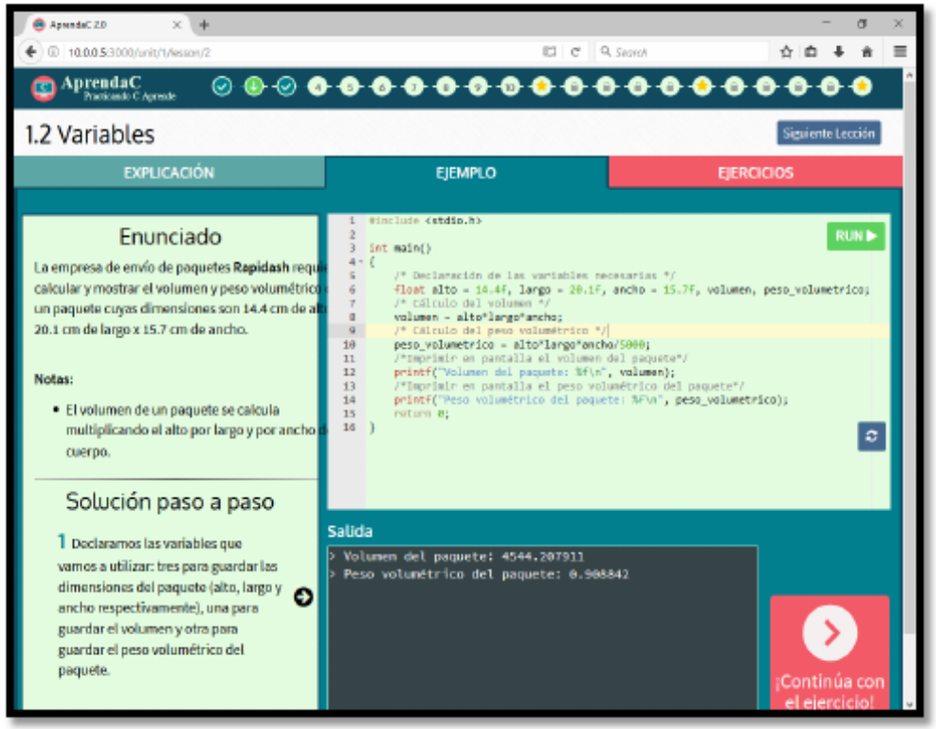

moz://a

EDITOR DE VÍDEO CON WASM



Aplica los efectos en tiempo real

APRENDE C (LEARN C)



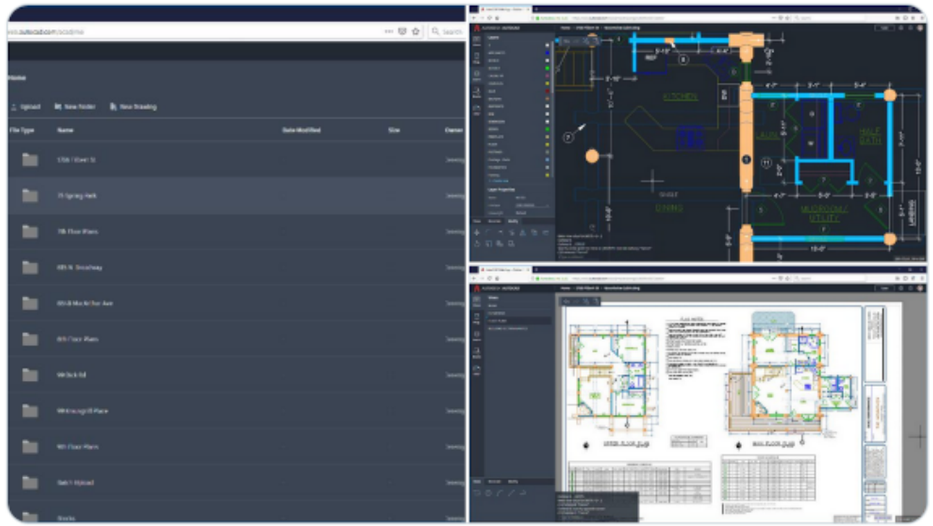
Un sitio para aprender C que se ejecuta en el navegador.

AUTOCAD EN LA WEB

AutoCAD
@AutoCAD

Starting today, you can use the @firefox browser to edit, create and view your CAD drawings. Learn more about the AutoCAD web app: autode.sk/2I35s2m

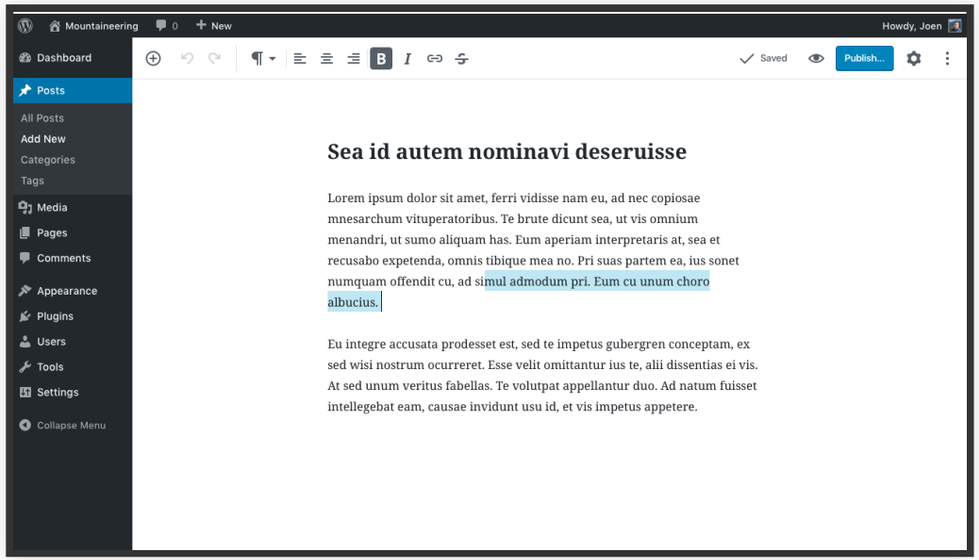
Traducir Tweet



6:42 p. m. · 7 jun. 2019 · Sprinklr

Fuente: Twitter.

INTÉRPRETE DE WORDPRESS GUTENBERG



El nuevo editor de WordPress utiliza un intérprete escrito en Rust.

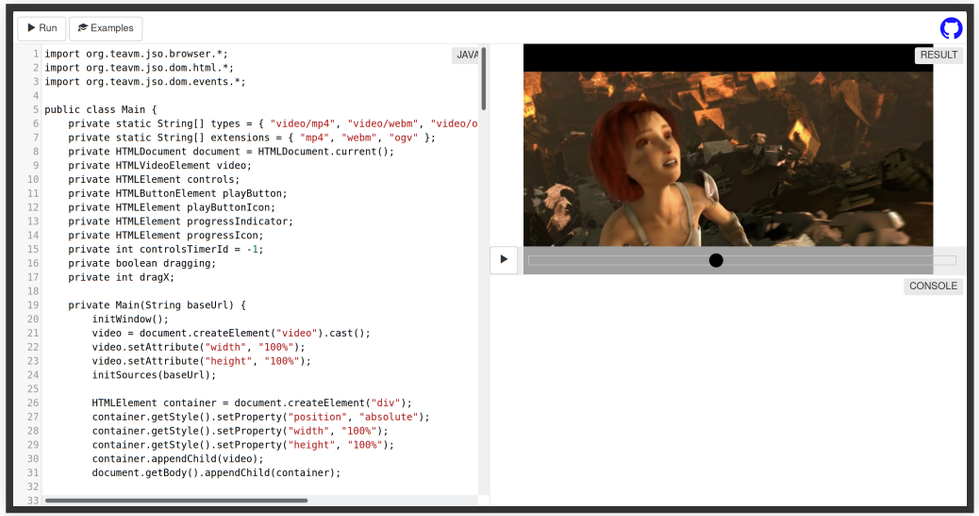
WASM binary

A yet-to-be-official benchmark is used to compare the performance of the actual Javascript parser against the Rust parser compiled as a WASM binary so that it can run in the browser. Here are the results:

file	Javascript parser (ms)	Rust parser as a WASM binary (ms)	speedup
`demo-post.html`	13.167	0.137	× 96
`shortcode-shortcomings.html`	26.784	0.225	× 119
`redesigning-chrome-desktop.html`	75.500	0.905	× 83
`web-at-maximum-fps.html`	88.118	0.698	× 126
`early-adopting-the-future.html`	201.011	0.927	× 217
`pygmalian-raw-html.html`	311.416	1.016	× 307
`moby-dick-parsed.html`	2,466.533	14.673	× 168

The WASM binary of the Rust parser is in average 159 times faster than the actual Javascript implementation. The median o
he speedup is 126.

TEA-VM DEMO: JAVA EN EL NAVEGADOR



Pruébalo en: http://teavm.org/sandbox/index.html

¡PODEMOS HACER FULL-STACK CON JAVA!

¡Adiós JS! (En parte)

¿FALTAN MAS EJEMPLOS DE JAVA?

¡TE INVITAMOS A CREARLOS!

FUTURAS CARACTERÍSTICAS DE WASM

- Multi-threading y Atómicas
- Manejo de excepciones
- Recolector de basura
- Variables globales mutables
- SIMD.
- JS BigInt y WebAssembly i64.

Mas en: https://webassembly.org/docs/future-features/

LIMITACIONES EN JAVA

- No existe un recolector de basura (GC).
- Soporte limitado de hilos.
- No hay un soporte oficial, solo proyectos de terceros.


¡NO ESPERES!
¡PRUÉBALO CON
WEBASSEMBLY STUDIO!

ENALCES RECOMENDADOS

- WASM official site
- WebAssembly Studio
- Web Docs information
- WASM Binary Structure
- Mozilla Hacks
- JWebAssembly
- TeaVM
- Bytecoder

GRACIAS POR SU ATENCIÓN

migueluseche.com

migueluseche@mozilla-hispano.org

@skatox | @mozillahispano