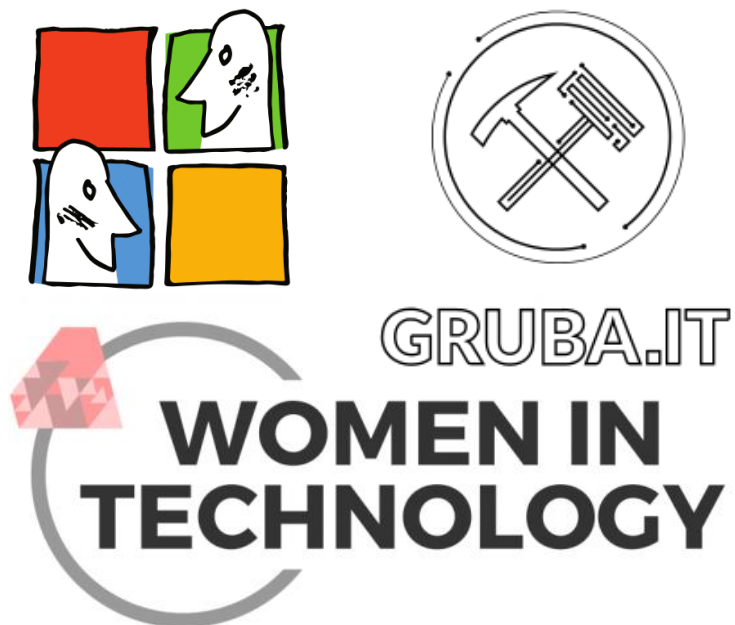


WebAssembly and .NET

Joanna Lamch





Who am I ?

Joanna Lamch

JLamch@gmail.com

JLamch.net

ProgramistkaKot.pl

Microsoft fangirl

Developer C#

.NET Framework 1.1

15 years (+ overtime)

Xamarin

SIENN

Community

Śląska Grupa Microsoft
Women In Technology
Gruba.IT





RATIONALTECHS.COM

Event Storming Praktycznie

**Rational Techs
wraz z SIENN**

zapraszają na

BEZPŁATNE
warsztaty

Gdzie?

Siedziba SIENN

**ul. Wagonowa 2
we Wrocławiu**

Kiedy?

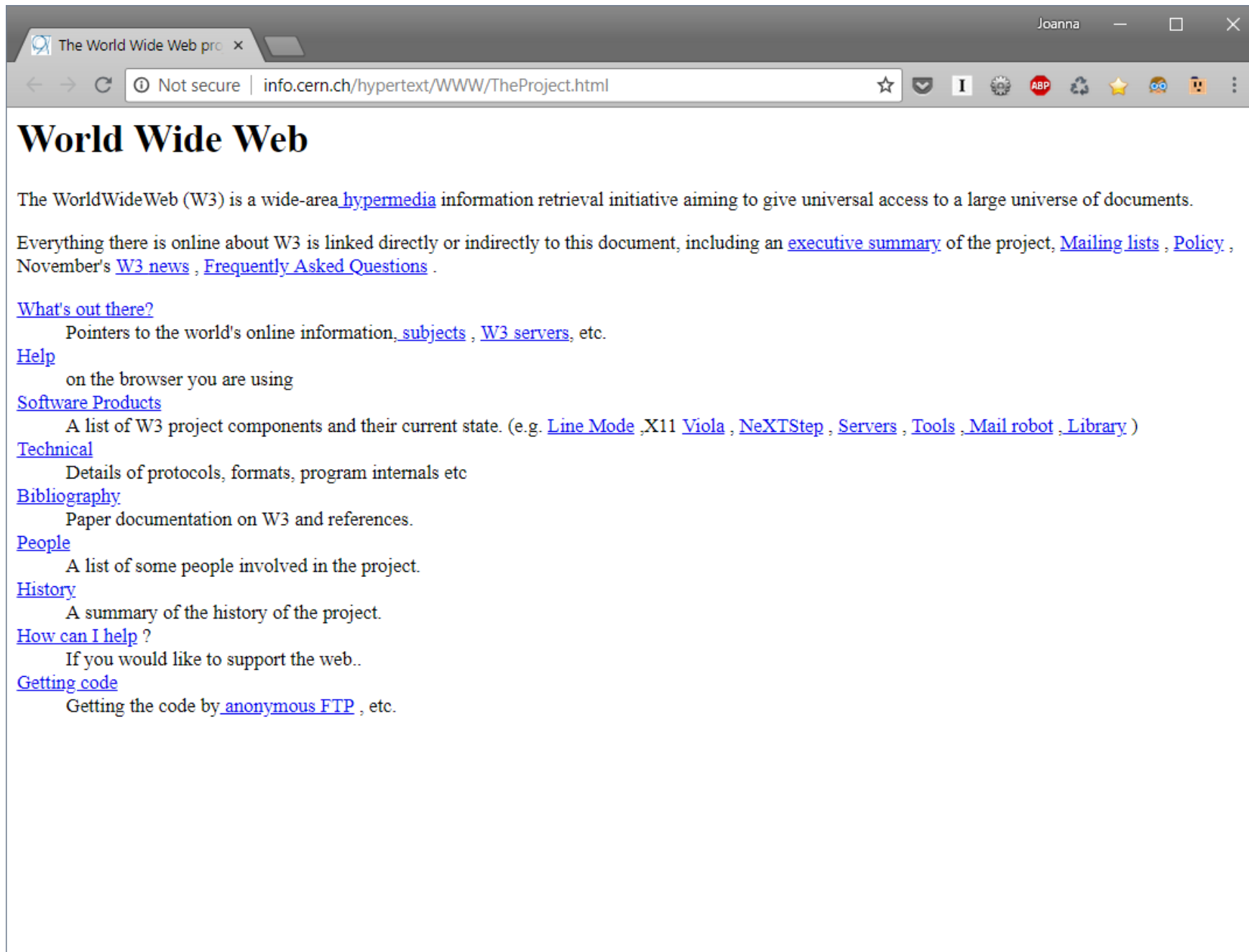
29.05 (środa)
17:45-20:15

→ Zapisy: rationaltechs@gmail.com

SIENN[®]

Back in the days...

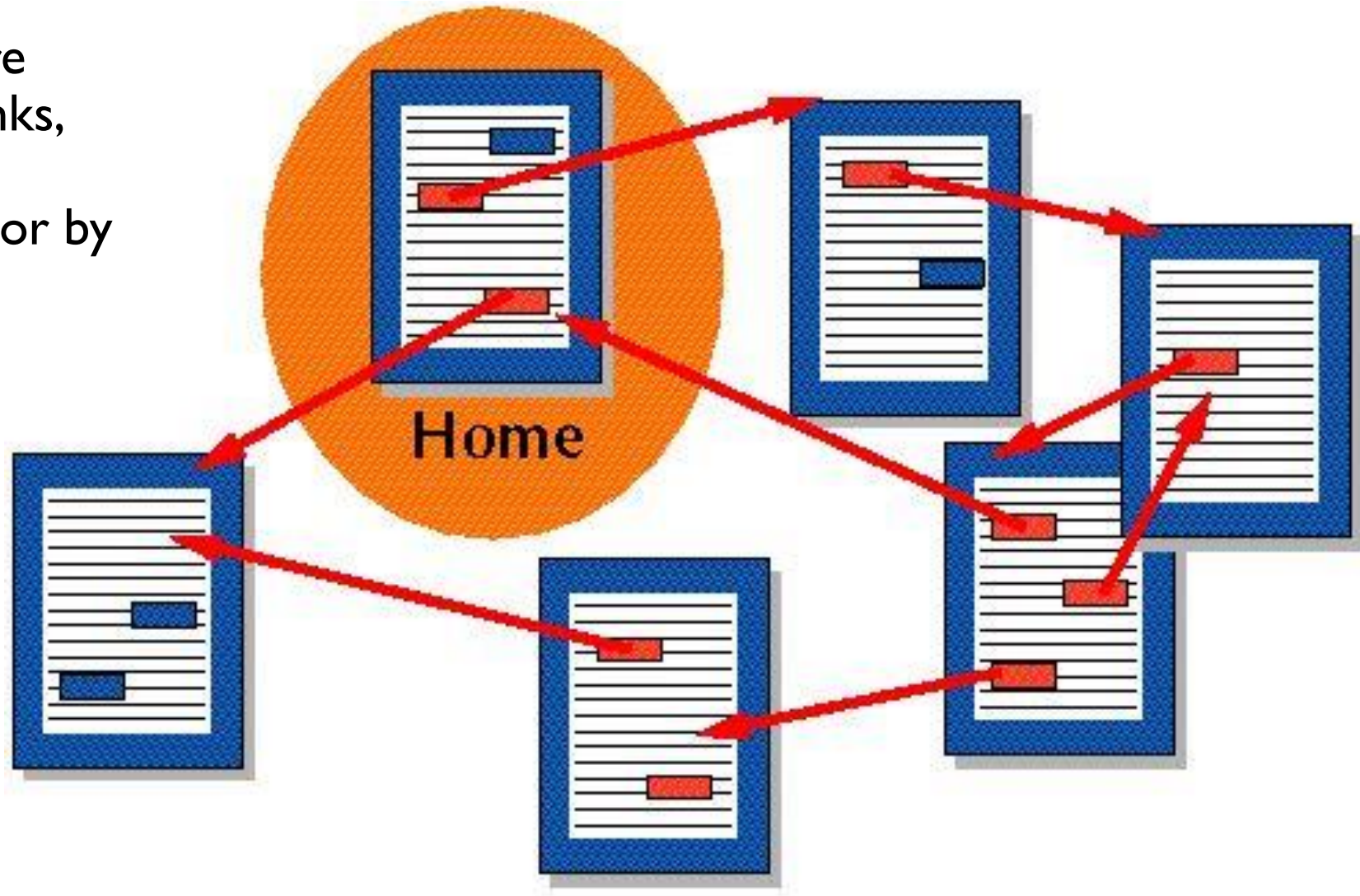




1991

Hypertext, HTML, HTTP

Hypertext documents are interconnected by hyperlinks, which are activated by a mouseClick, keypress set or by touching the screen



JavaScript

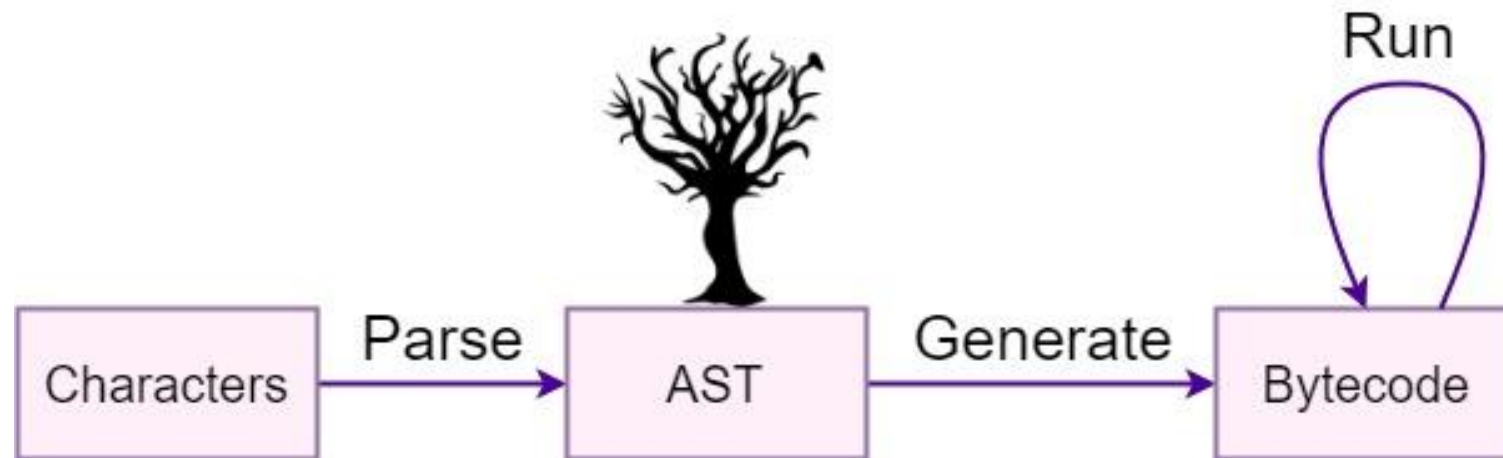


created in 10 days in May 1995, by Brendan Eich for Netscape

Adding interactivity to HTML pages

1995

JavaScript



1995

JavaScript



1995

Other plugins

Java Applets [1997]

ActiveX [1996]

Flash [1996]

Silverlight [2007]

1996+



Other plugins

Java Applets [1997]

ActiveX [1996]

Flash [1996]

Silverlight [2007]

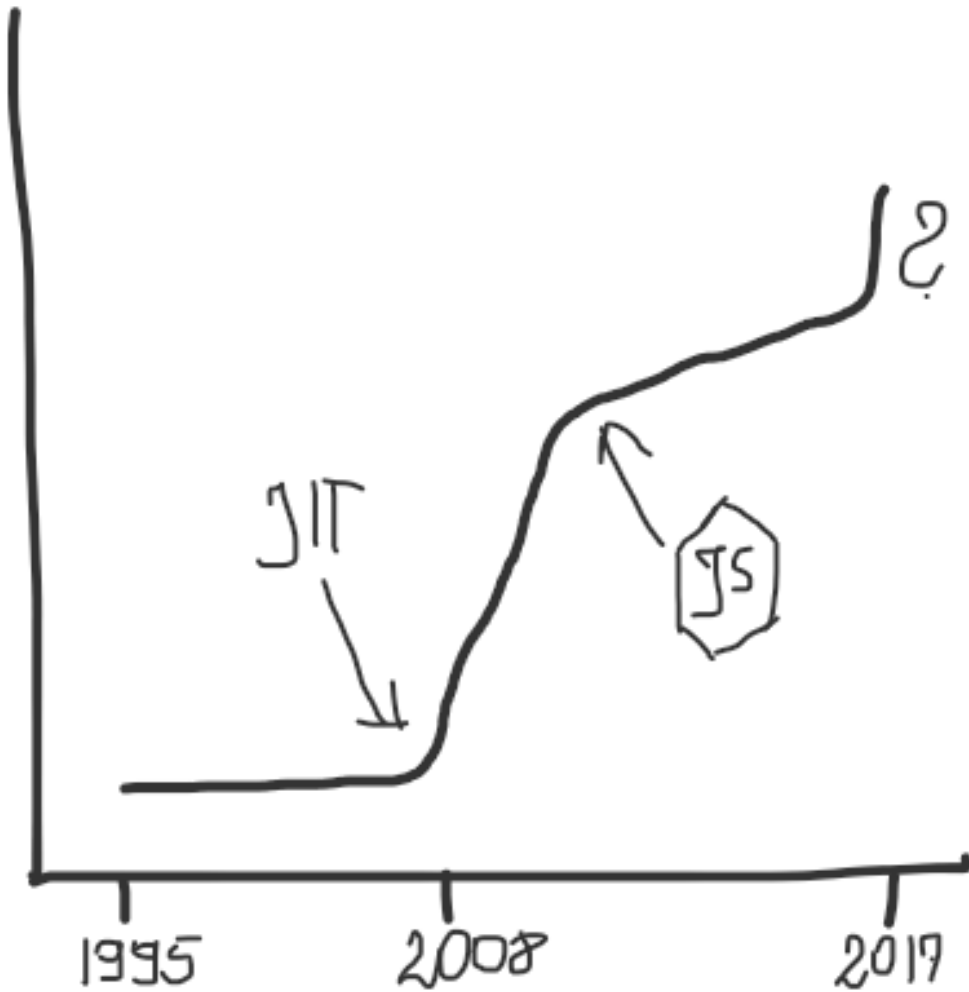
1996+



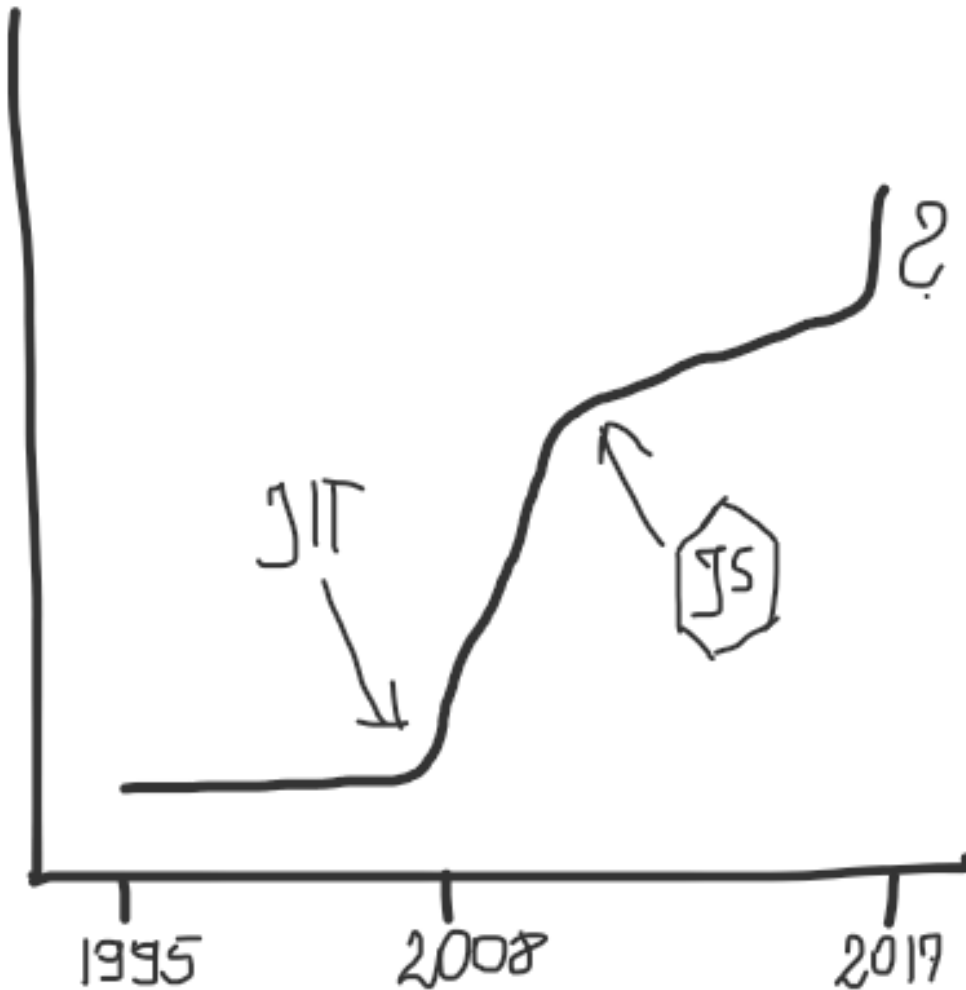
All of them are deprecated
or will be deprecated soon



Browser performance wars (2008+)

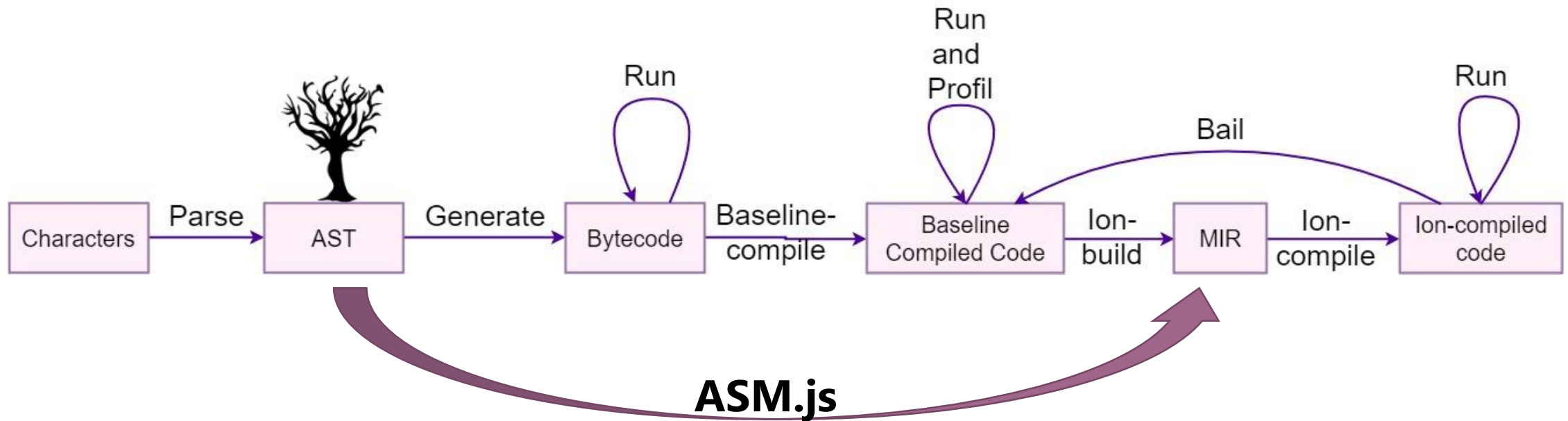


Browser performance wars (2008+)

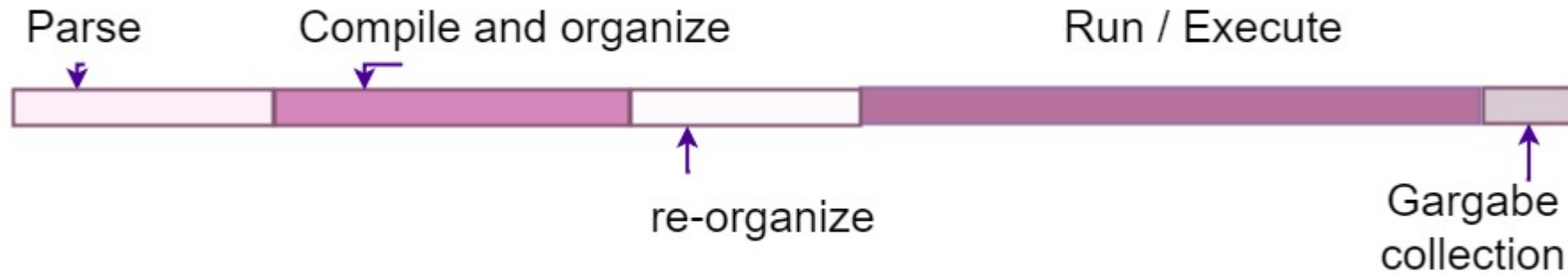


moz://a

Browser performance wars (2008+)



Browser performance wars (2008+)



Today's JavaScript



Different way the same language

there is a LOT of JavaScript

JavaScript is the language of the Web

But it's not very good Assembly Language
(still human readable simple language)

Today's JavaScript

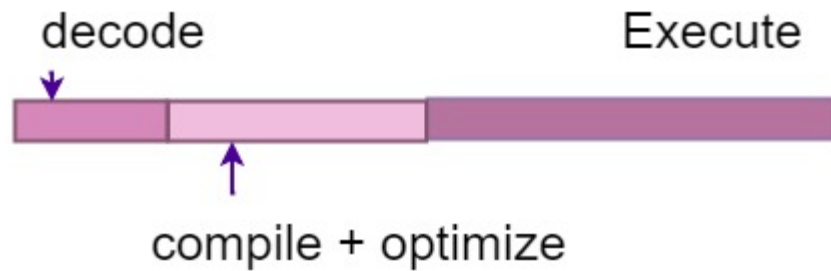
JS



WebAssembly What? Why?

A new low-level **binary** format **for the web** (WASM)

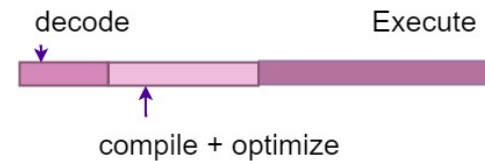
It's a bytecode for web / compilation target $\Rightarrow$ **maximized performance**



WebAssembly What? Why?

A new low-level **binary** format **for the web** (WASM)

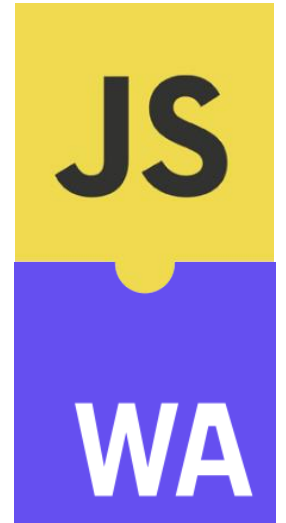
It's a bytecode for web / compilation target ➡ **maximized performance**



WebAssembly **is not designed to replace** JS, but to coexist

Sandboxed runtime in JS virtual machine

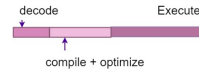
Security it runs locally in JSVM



WebAssembly What? Why?

A new low-level **binary** format **for the web** (WASM)

It's a **bytecode** for web / compilation target => **maximized performance**

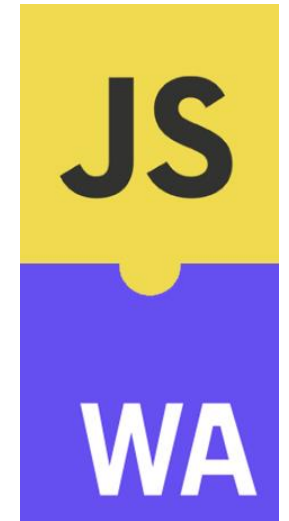


WebAssembly **is not designed to replace** JS, but to coexist

Sandboxed runtime in JS virtual machine

Security it runs locally in JSVM

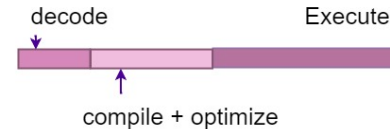
Supported in **all** big browsers
– W3C open specification



WebAssembly What? Why?

A new low-level **binary** format **for the web** (WASM)

It's a bytecode for web / compilation target ➡ **maximized performance**



WebAssembly **is not designed to replace** JS, but to coexist

Sandboxed runtime in JS virtual machine

Security it runs locally in JS VM

Supported in **all** big browsers – W3C open specification

Compiled from **other languages**



WASM opens possibilities for other languages



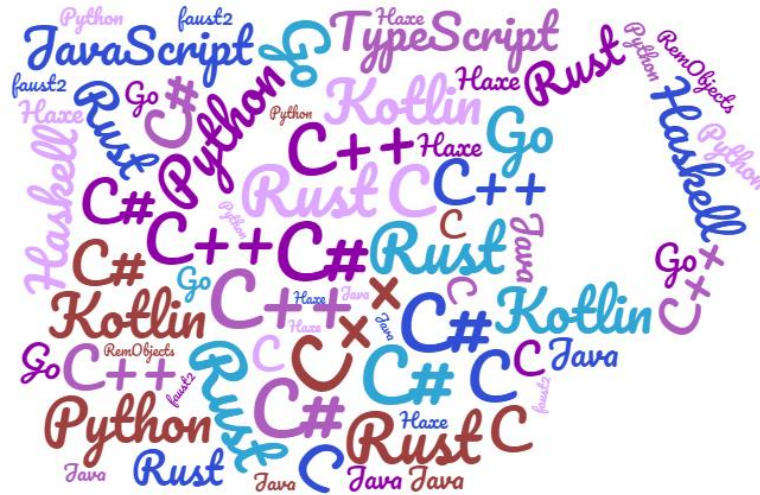
WASM opens possibilities for other languages

Rust

C++

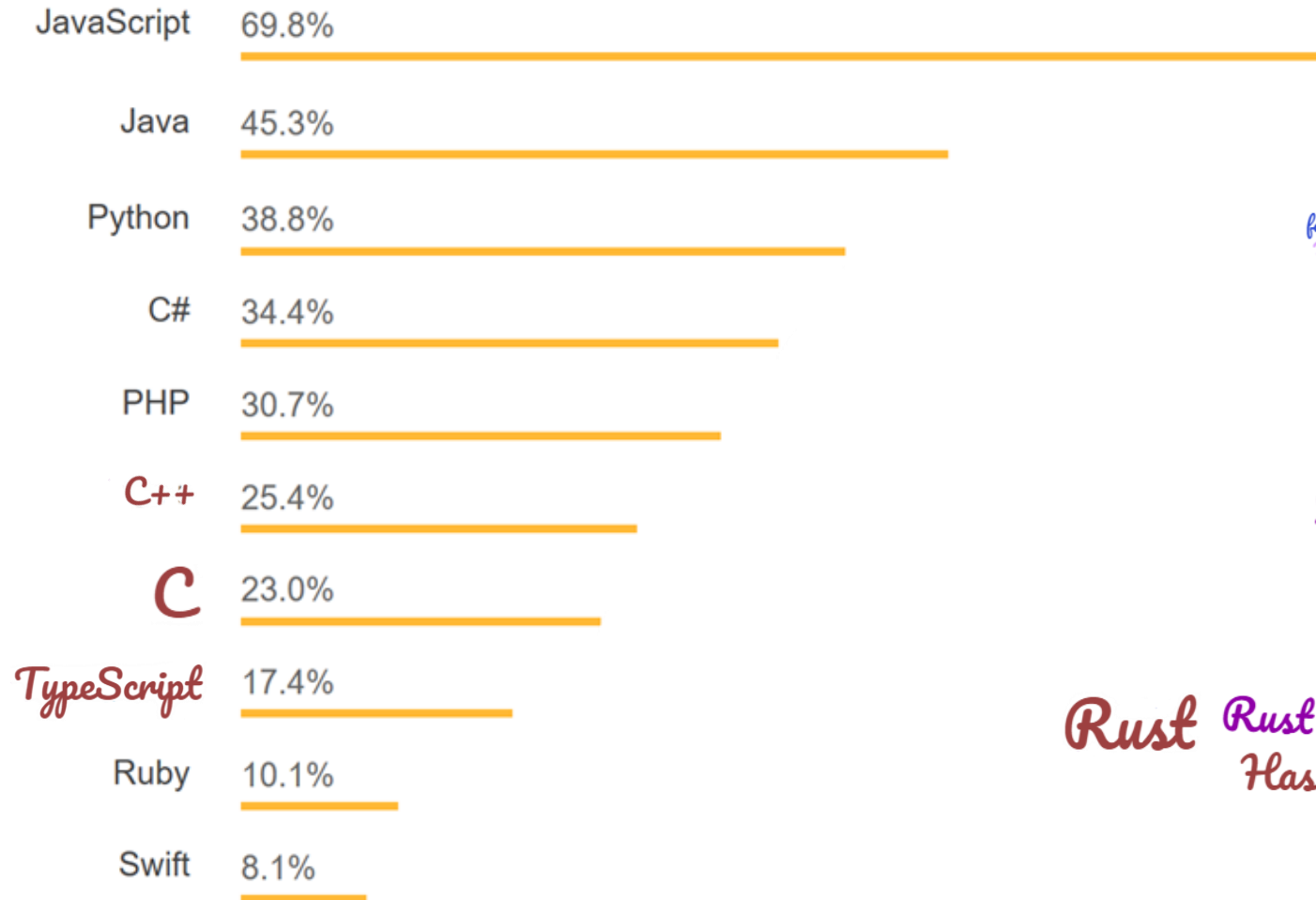
C

TypeScript



Rust JavaScript Kotlin C# Kotlin
Haskell Go Kotlin TypeScript Go C# Python
RemObjects RemObjects

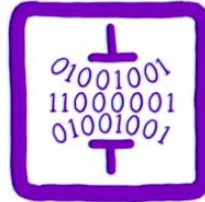
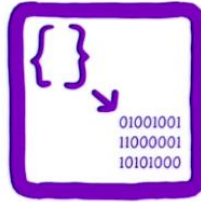
WASM opens possibilities for other languages



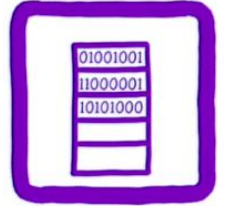
WebAssembly current state: MVP

- **Compilation target**

Binary so compact



- **Linear memory**



- **4 types, 67 instructions, stack machine**

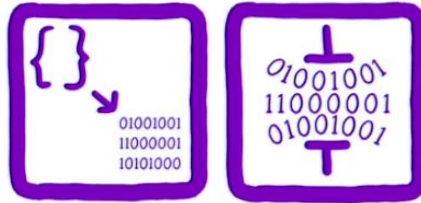
- **Fast execution**



WebAssembly current state: MVP

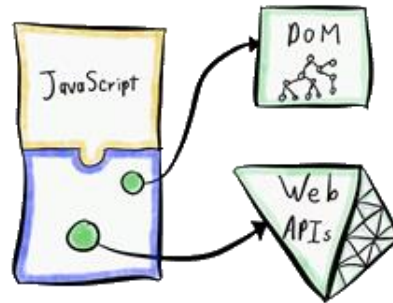
- Compilation target

Binary so compact

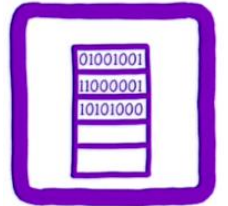


- 4 types, 67 instructions, stack machine

- No Web APIs, DOM access/manipulation



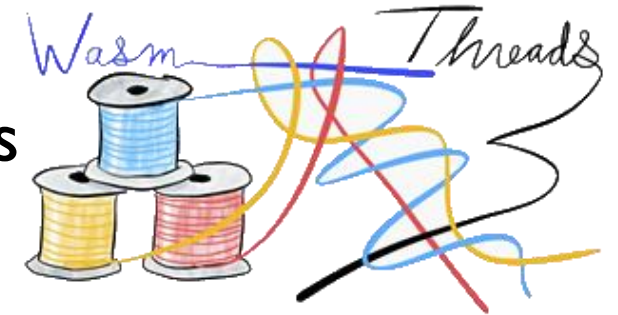
- Linear memory



- Fast execution
or not so fast
not so fast load



- no Threads



WebAssembly How?



emscripten

executes the **Emscripten** compiler

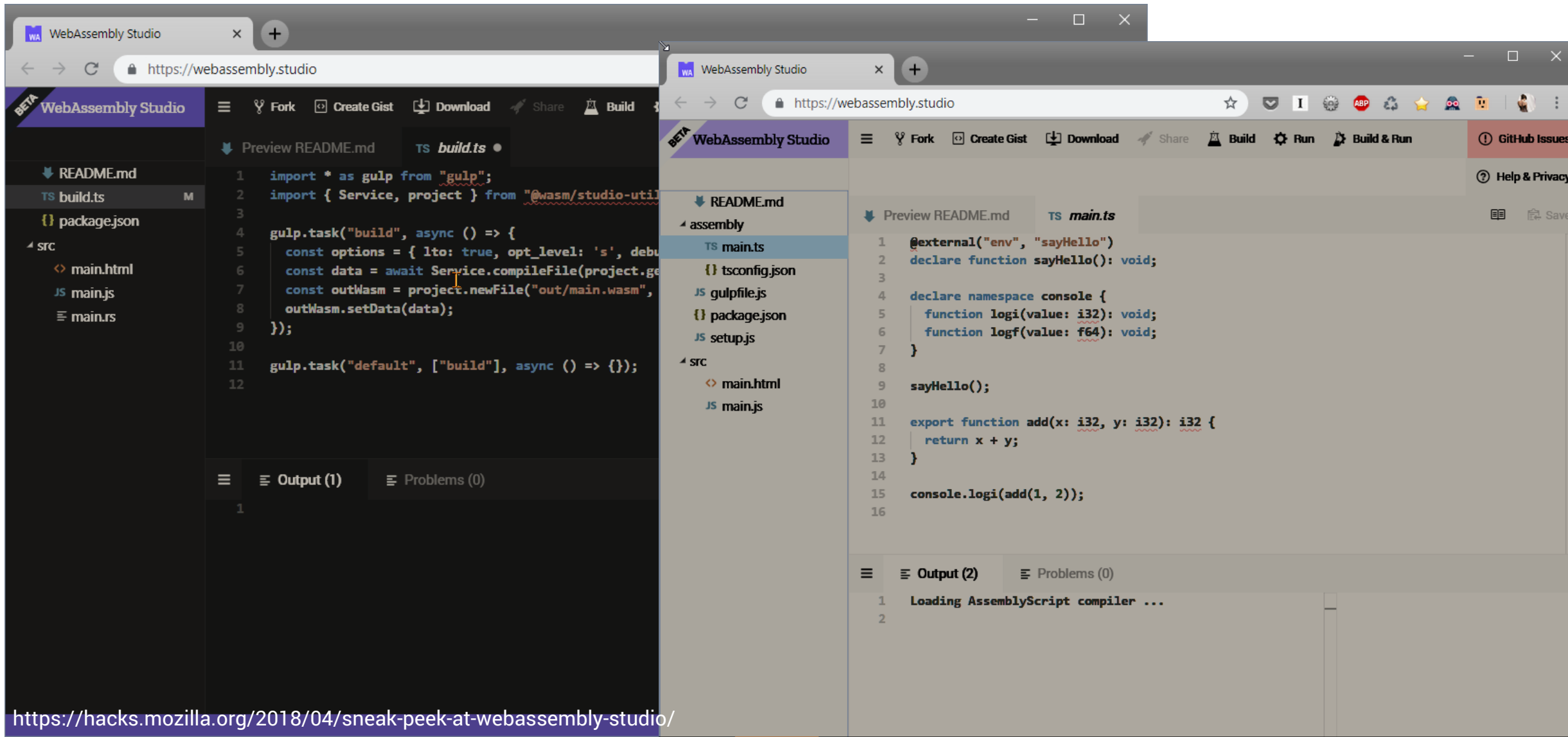
name our output file

```
$emcc -s WASM=1 -o example.js example.cpp
```

Emscripten outputs WebAssembly - switch

input file

WebAssembly studio C/Rust/TypeScript



Demo

<http://mbebenita.github.io/WasmExplorer/>

<https://wasdk.github.io/WasmFiddle/>

<https://webassembly.studio/>



WasmExplorer

C++11 -Os

COMPILE

1 float add(float a, float b){

2 return a+b;

3 }

Firefox x86 Assembly

>

▼ wasm-function[0]:

sub rsp, 8; 0x000000 48 83 ec 08

addss xmm0, xmm1; 0x000004 f3 0f 58 c1

nop; 0x000008 66 90

add rsp, 8; 0x00000a 48 83 c4 08

ret; 0x00000e c3

Wat

ASSEMBLE

DOWNLOAD

1 (module

2 (table 0 anyfunc)

3 (memory \$0 1)

4 (export "memory" (memory \$0))

5 (export "_Z3addff" (func \$_Z3addff))

6 (func \$_Z3addff (; 0 ;) (param \$0 f32) (param

7 \$1 f32) (result f32)

8 (f32.add

9 (get_local \$0)

10 (get_local \$1)

11)

12)

13)

<https://bit.ly/2JkqFYt>

WasmExplorer

Firefox x86 Assembly



▼ `wasm-function[0]:`

<code>sub</code>	<code>rsp, 8</code>	<code>; 0x000000 48 83 ec 08</code>
<code>addss</code>	<code>xmm0, xmm1</code>	<code>; 0x000004 f3 0f 58 c1</code>
<code>nop</code>		<code>; 0x000008 66 90</code>
<code>add</code>	<code>rsp, 8</code>	<code>; 0x00000a 48 83 c4 08</code>
<code>ret</code>		<code>; 0x00000e c3</code>

BETA

WebAssembly Studio

Fork

Create Gist

Download

Share

Build

Run

Build & Run

Help & Privacy

GitHub Issues

main.c

main.js

main.wasm

Save

Save

README.md

build.ts

package.json

src

main.c

main.html

main.js

out

main.wasm

1

2

3

4

5

6

7

8

9

fetch('../out/main.wasm').then(response =>

response.arrayBuffer()

)

.then(bytes => WebAssembly.instantiate(bytes)).then(result=> {

instance= result.instance;

document.getElementById("container").textContent = "Result: "

+ instance.exports.add(19.19, 23.23);

}).catch(console.error);

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

#include <stdio.h>

#include <sys/uio.h>

#include <math.h>

#define WASM_EXPORT __attribute__((visibility("default")))

WASM_EXPORT

int main() {

return 42;

}

WASM_EXPORT

double add(float a, float b){

float f=a+b;

float num = floor(100*f)/100;

return f;

}

Output (50)

Problems (0)

41

42

43

44

45

46

47

48

49

50

[info]: Task build is completed

[info]: Task build is running...

[info]: Task build is completed

[info]: Task build is running...

[info]: Task build is completed

Downloading Project ...

Project Zip CREATED

[info]: Task build is running...

[info]: Task build is completed

Result: 42.41999816894531

- README.md
- build.ts
- package.json
- src
- C

main.c
- main.html
- main.js
- out
- main.wasm

C main.c

Save

```
1  #include <stdio.h>
2  #include <sys/uio.h>
3  #include <math.h>
4  #define WASM_EXPORT __attribute__((visibility("default")))
5
6  WASM_EXPORT
7  int main() {
8      return 42;
9  }
10
11  WASM_EXPORT
12  double add(float a, float b){
13      float f=a+b;
14      float num = floor(100*f)/100;
15      return f;
16  }
```

bility("default"))))

099816894531

50

📄 README.md

TS build.ts

{ } package.json

src

C main.c

<> main.html

JS main.js

out

main.wasm

```
1 fetch('../out/main.wasm').then(response =>
2   response.arrayBuffer()
3 )
4 .then(bytes => WebAssembly.instantiate(bytes)).then(result=> {
5   instance= result.instance;
6   document.getElementById("container").textContent = "Result: "
7   + instance.exports.add(19.19, 23.23);
8 }).catch(console.error);
9
```

Output (50)Problem

41 [info]: Task build is com

42 [info]: Task build is run

43 [info]: Task build is com

44 [info]: Task build is run

45 [info]: Task build is com

46 Downloading Project ...

47 Project Zip CREATED

48 [info]: Task build is running...

49 [info]: Task build is completed

50

Result: 42.41999816894531

Result: 42.41999816894531

BETA

WebAssembly Studio

Fork

Create Gist

Download

Share

Build

Run

Build & Run

Help & Privacy

GitHub Issues

main.c

1

fetch

2

res

3

)

4

.then

5

ins

6

doc

7

+ i

8

}).ca

9

main.c

#include <stdio.h>

#include <sys/uio.h>

#include <math.h>

#define WASM_EXPORT __attribute__((visibility("default")))

WASM_EXPORT

int main() {

return 42;

}

WASM_EXPORT

double add(float a, float b){

float f=a+b;

float num = floor(100*f)/100;

return f;

}

Output

41

[info

42

[info

43

[info

44

[info

45

[info

46

Downl

47

Proje

48

[info

49

[info

50

src

main.c

main.html

main.js

out

main.wasm

Result: 42.41999816894531

JS main.js

 [main.wasm](#)

C main.c

TS build.ts

```
{}
```

package.json

src

C main.c

<> main.html

JS main.js

◀ out

main.wasm

```
1 fetch('../out/main.wasm').then(response =>
2   response.arrayBuffer()
3 )
4 .then(bytes => WebAssembly.instantiate(bytes)).then(result=> {
5   instance= result.instance;
6   document.getElementById("container").textContent = "Result: "
7   + instance.exports.add(19.19, 23.23);
8 }).catch(console.error);
9
```

```
1 #include <stdio.h>
2 #include <sys/uio.h>
3 #include <math.h>
4 #define WASM_EXPORT __attribute__((visibility("default")))
5
6 WASM_EXPORT
7 int main() {
8     return 42;
9 }
10
11 WASM_EXPORT
12 double add(float a, float b){
13     float f=a+b;
14     float num = floor(100*f)/100;
15     return f;
16 }
17
18
```

≡ Output (50)

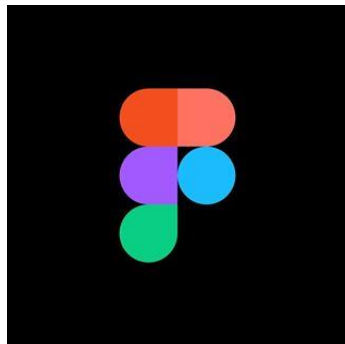
Problems (0)

```
41 [info]: Task build is completed
42 [info]: Task build is running...
43 [info]: Task build is completed
44 [info]: Task build is running...
45 [info]: Task build is completed
46 Downloading Project ...
47 Project Zip CREATED
48 [info]: Task build is running...
49 [info]: Task build is completed
50
```

Result: 42.41999816894531



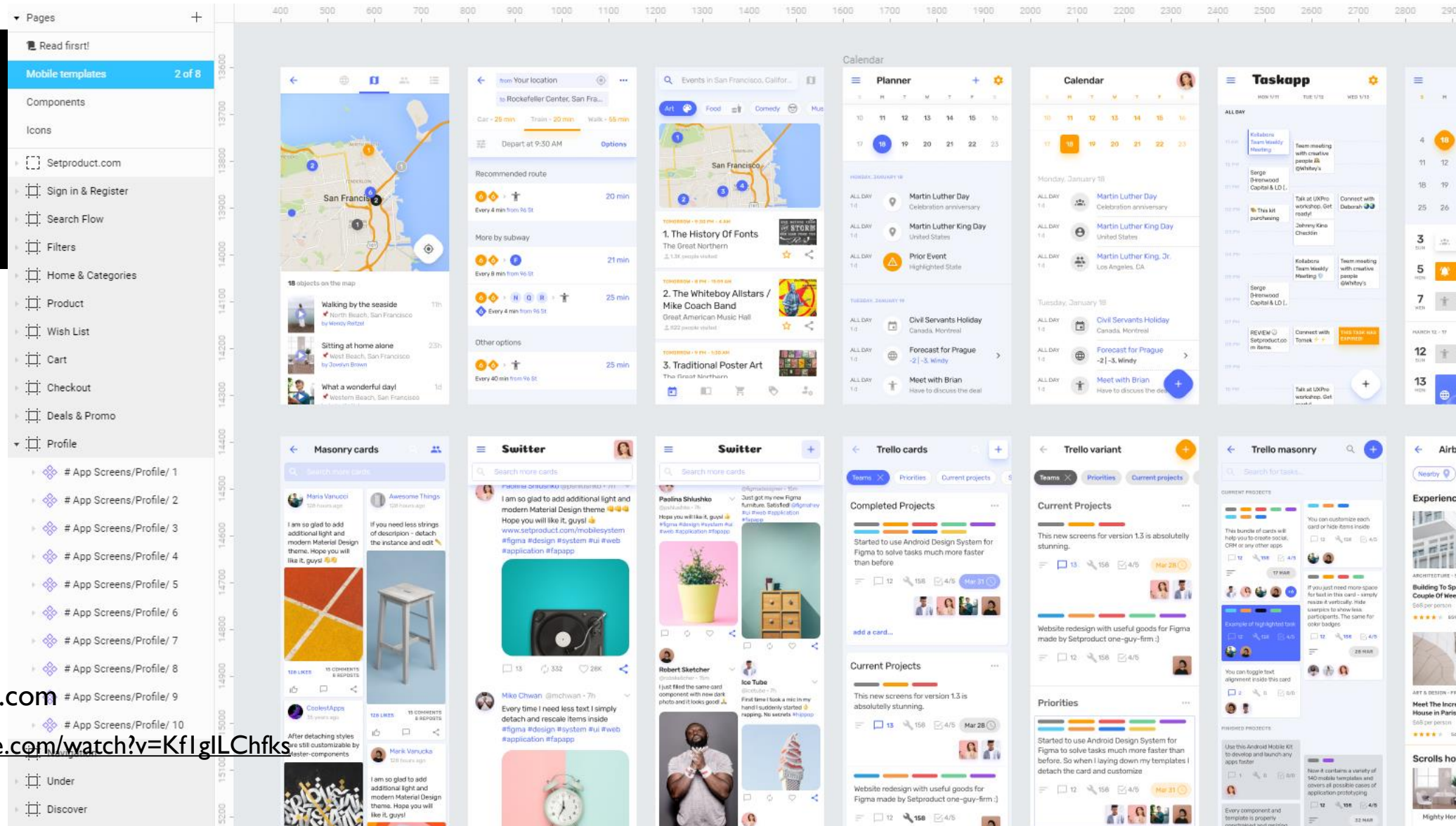
WASM in real world



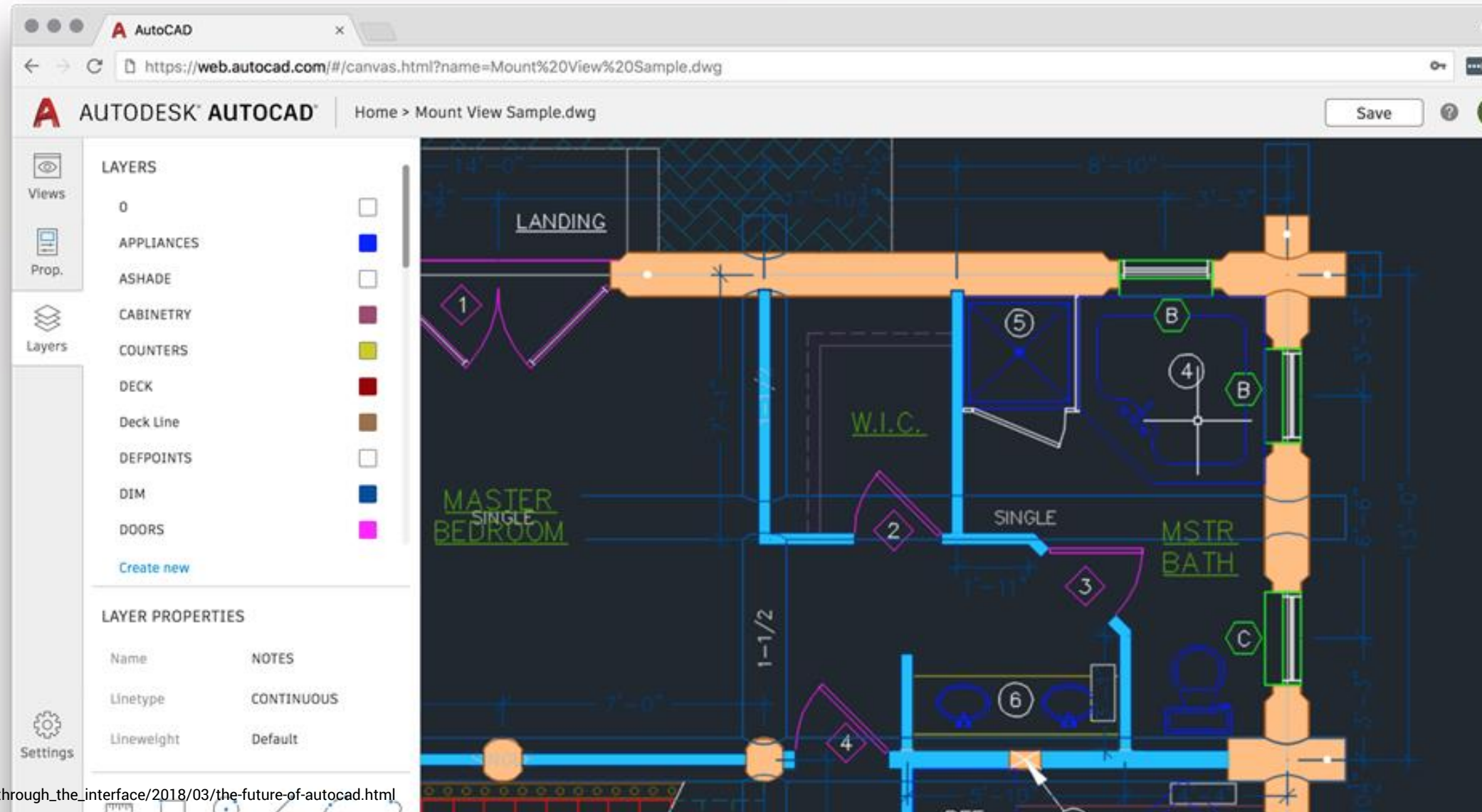
Figma

<https://www.figma.com>

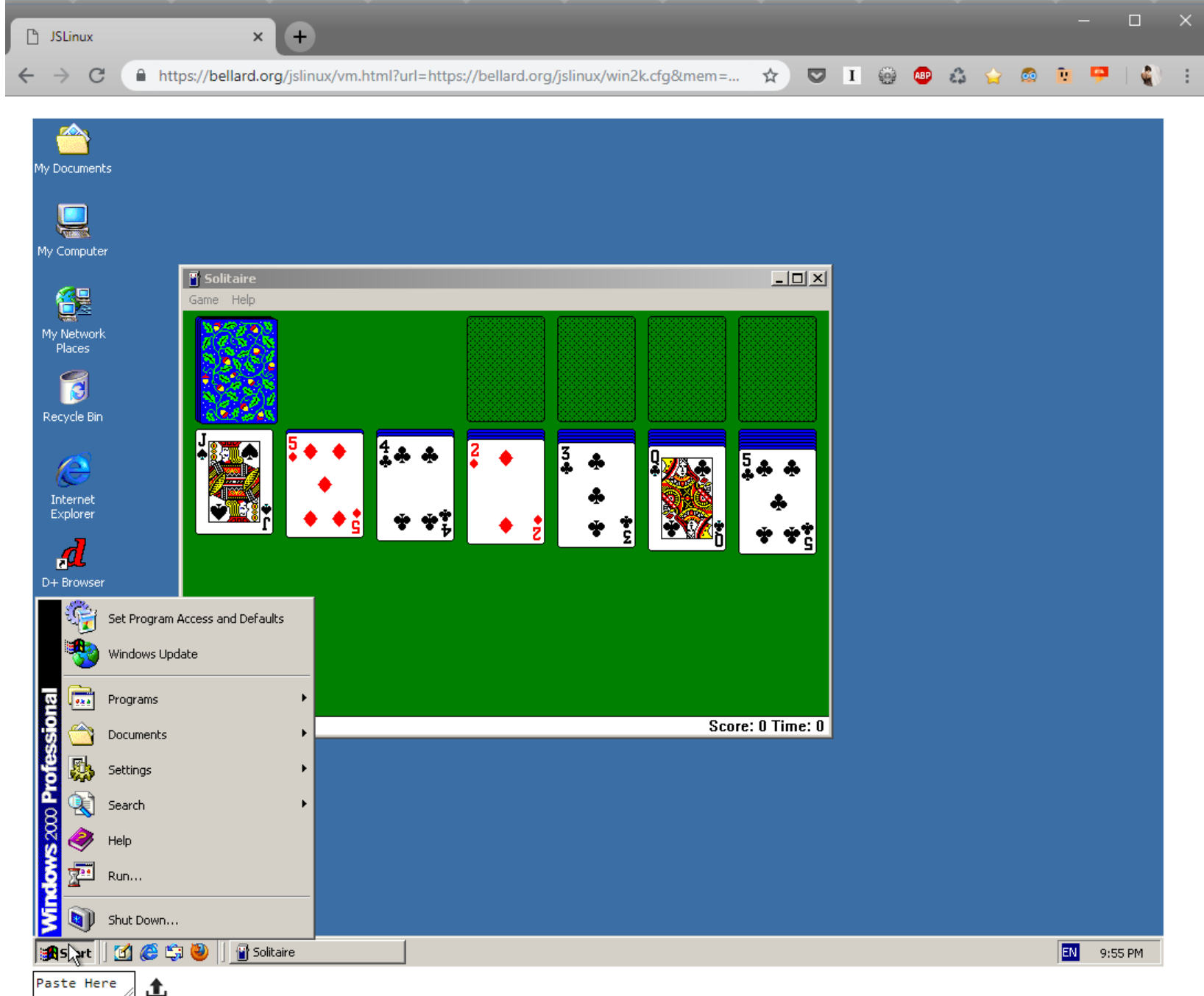
<https://www.youtube.com/watch?v=KfIglLChfks>



WASM in real world

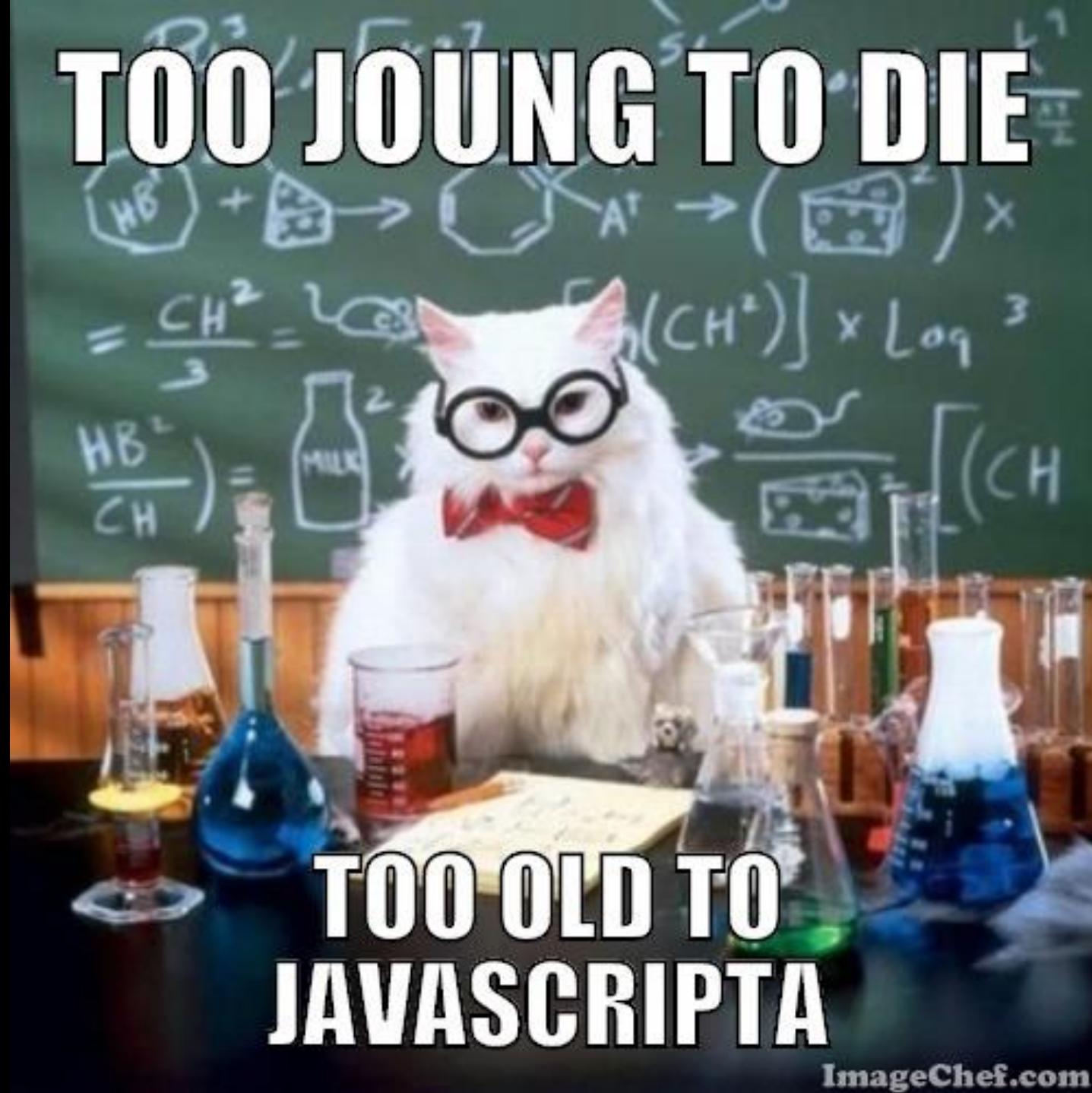








TOO YOUNG TO DIE



**TOO OLD TO
JAVASCRIPTA**



Why use .NET for browser apps?

Stable, mature, productive:

.NET Standard, MSBuild

Fast, scalable, reliable:

.NET Core for backend services

Modern languages:

Innovations in C#, F#, Razor

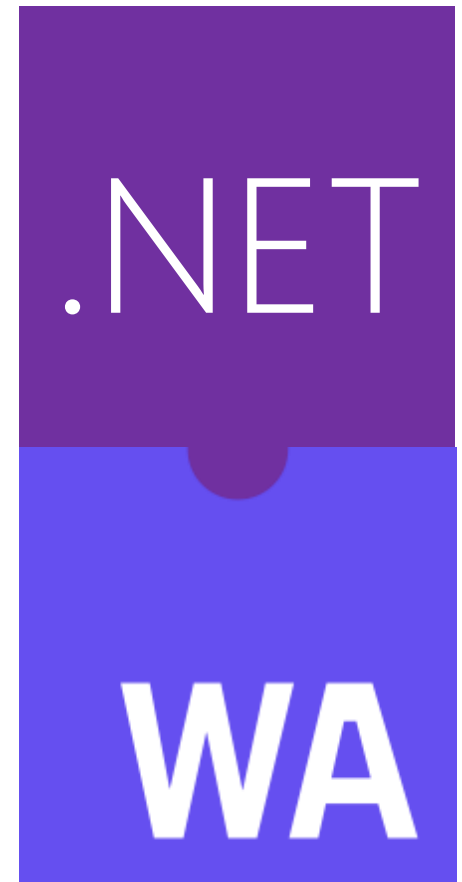
First-rate dev tools:

Visual Studio, IntelliSense

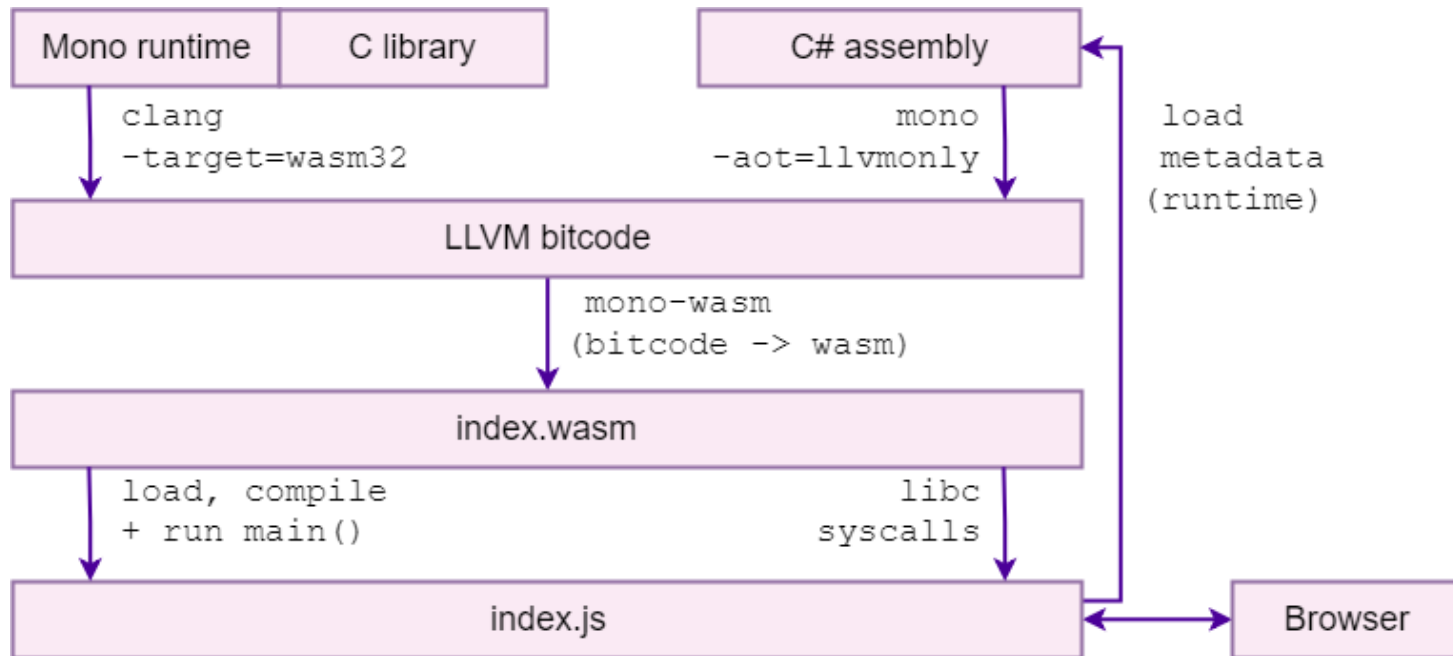
Solid foundation to build on

Leverage existing skills and knowledge

Because We can 



C# in wasm ? Mono !



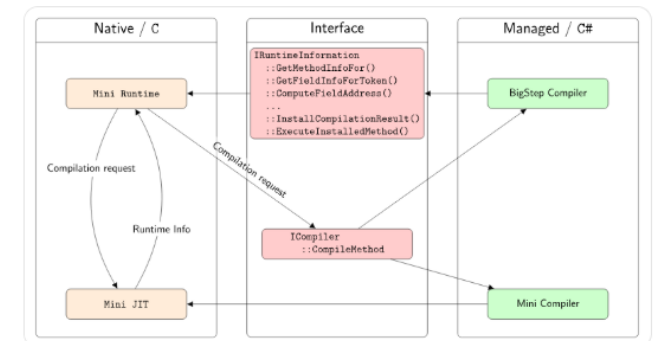
Miguel de Icaza
@migueldeicaza

Following

Mono now allows the code generation engine (for JIT and AOT) to be replaced. Not only replaced, but replaced with C# or F# code!

No more low-level, error-prone, unsafe C code!

Details here: [mono-project.com/news/2018/09/11/...](https://www.mono-project.com/news/2018/09/11/csharp-jit/)



3:19 PM - 11 Sep 2018

<https://github.com/migueldeicaza/mono-wasm>

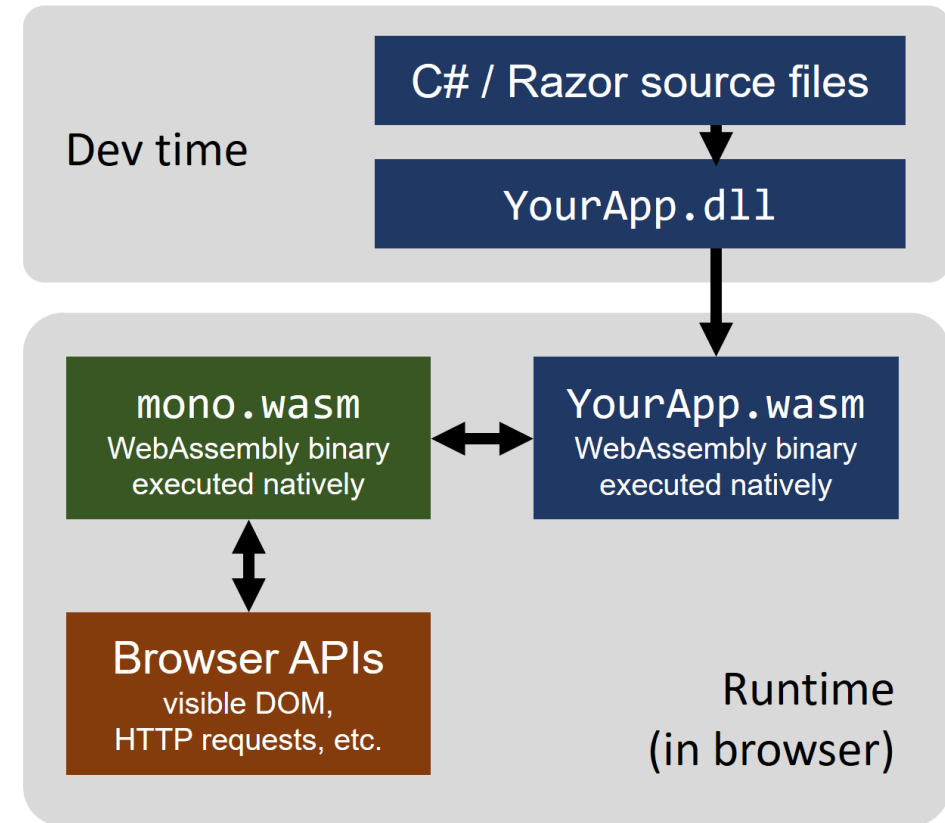
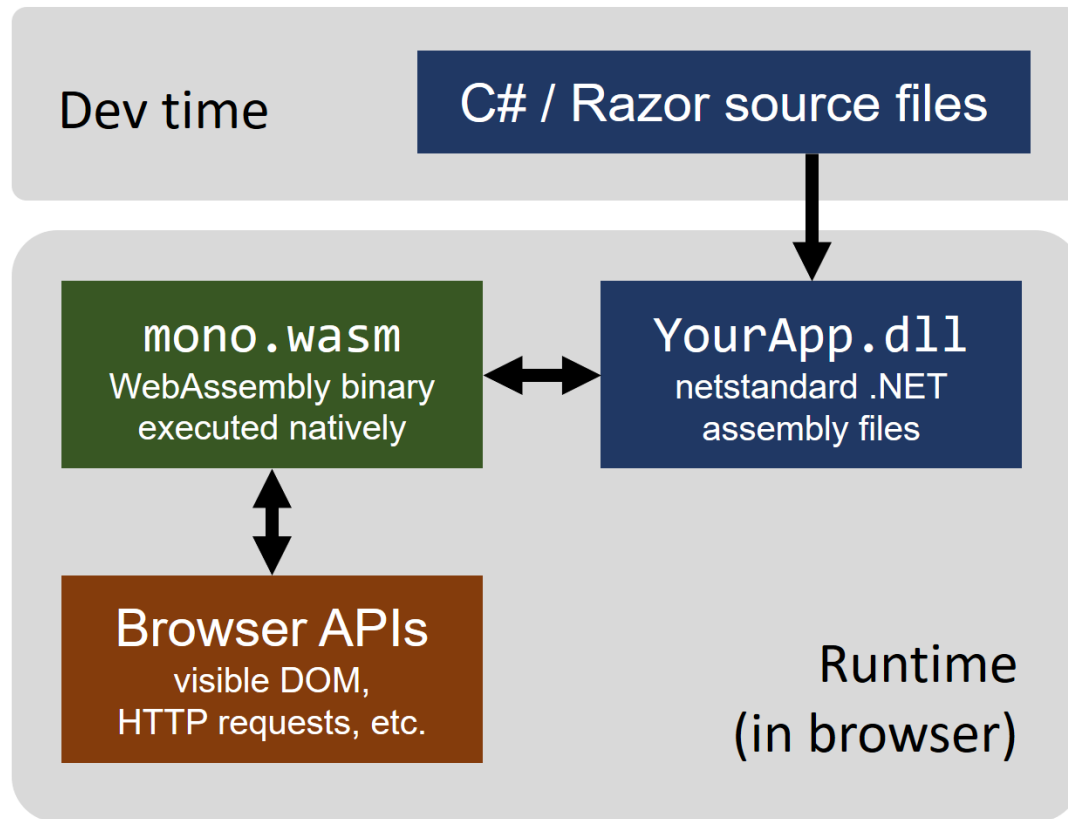
<https://www.mono-project.com/news/2018/09/11/csharp-jit/>

<https://twitter.com/migueldeicaza/status/1039639597435641856>

Interpreter mode

vs

AOT compiled



XAML for desktop app



XAML standard is ready

XAML for desktop app



XAML standard is ready and dead

XAML for desktop app



XAML standard is ready and dead

- Xamarin.Forms
 - UWP
 - Silverlight
 - WPF
- 
- A purple curly brace grouping the last three items in the list: Silverlight, WPF, and the item below it (which is not visible in the image).

Ooui



C# Platform Specific

iOS Renderers



C# Platform Specific

Android Renderers



C# Platform Specific

Windows Renderers



Web Platform Specific

Web Renderers

Shared C# UI Code
(Xamarin.Forms XAML)

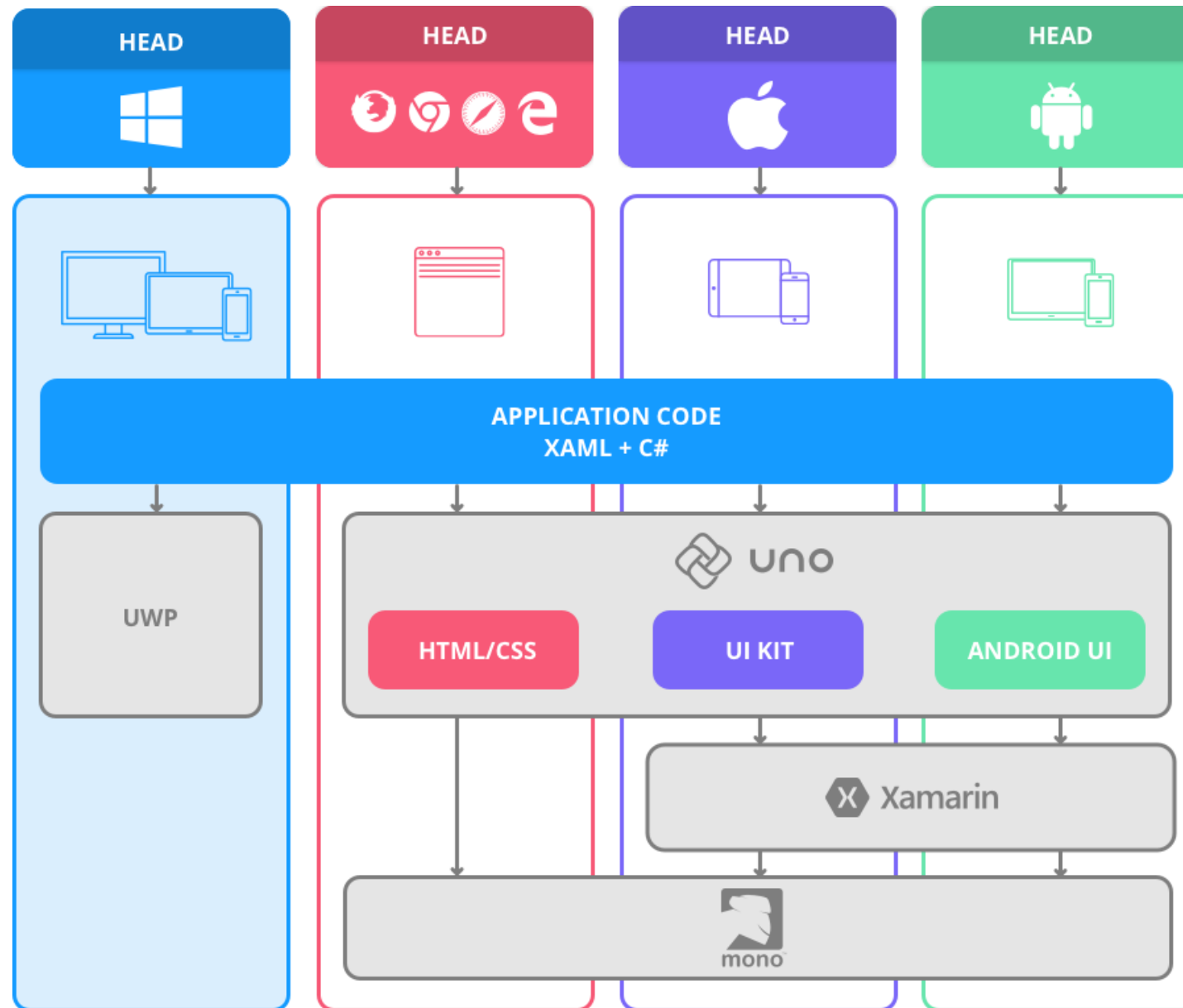
Shared C# App Logic
(ViewModels, Models etc.)

DLLs
(external packages & libraries)

Mono

.NET

UNO



FrogUI



<https://www.lesarde.com/>



Blazor? Razor components ?

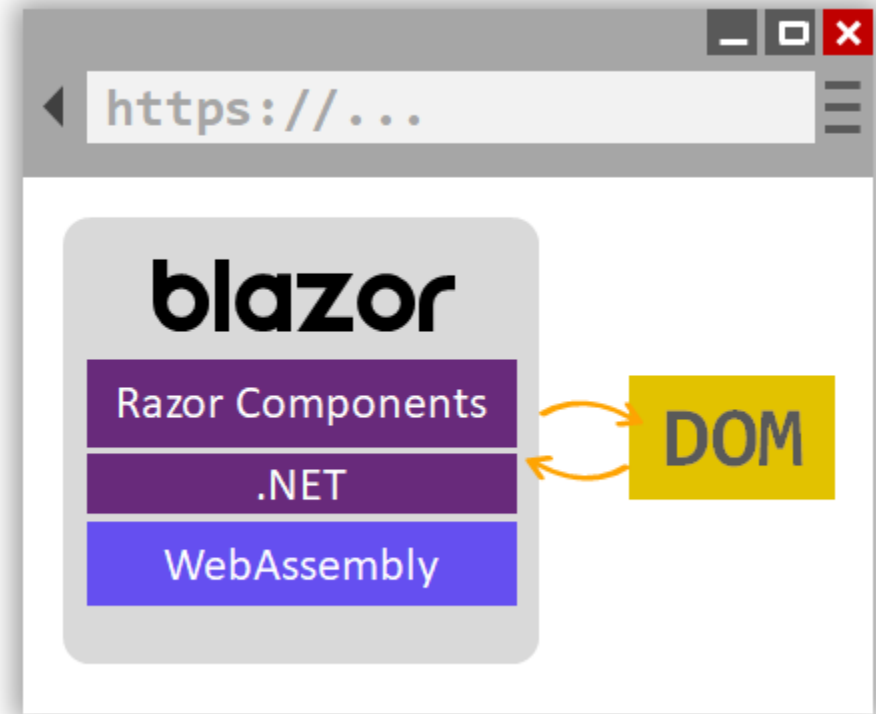
Blazor

For information about using Blazor, see blazor.net.

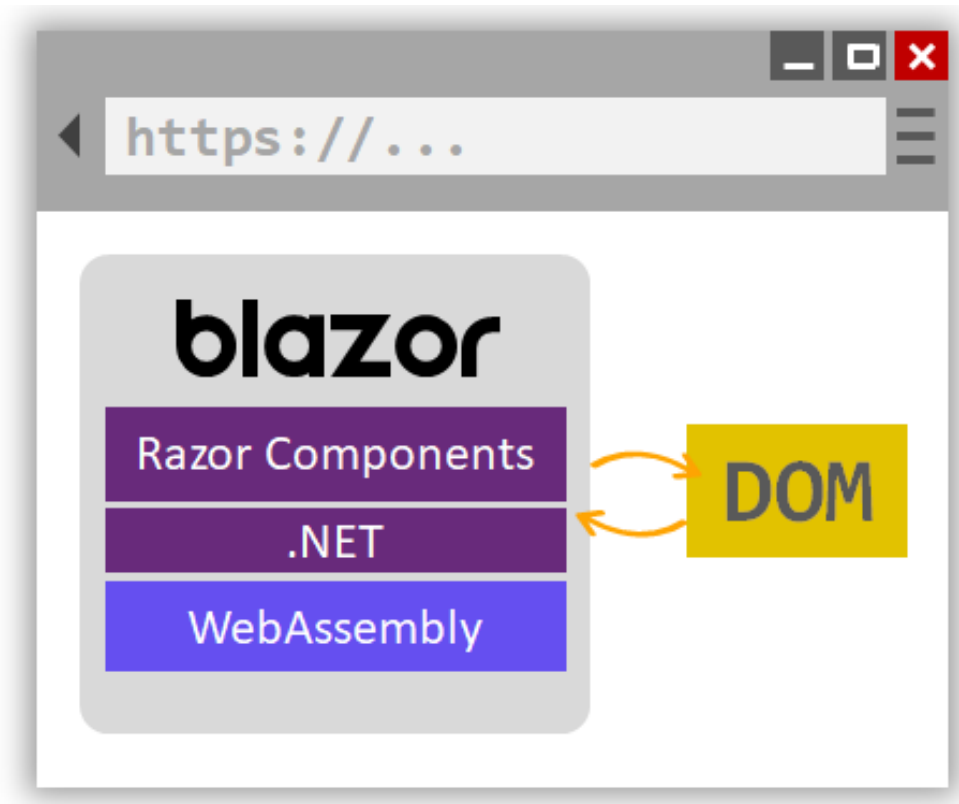
Sources moved

Almost all the sources for Blazor and the Razor Components programming model have moved [here in the central ASP.NET Core repo](#). We are also in the process of migrating open issues from here to there.

This is in preparation for shipping Razor Components as a built-in feature of ASP.NET Core 3.0. *Note: client-side Blazor remains experimental while we continue to work on making the WebAssembly runtime complete.*



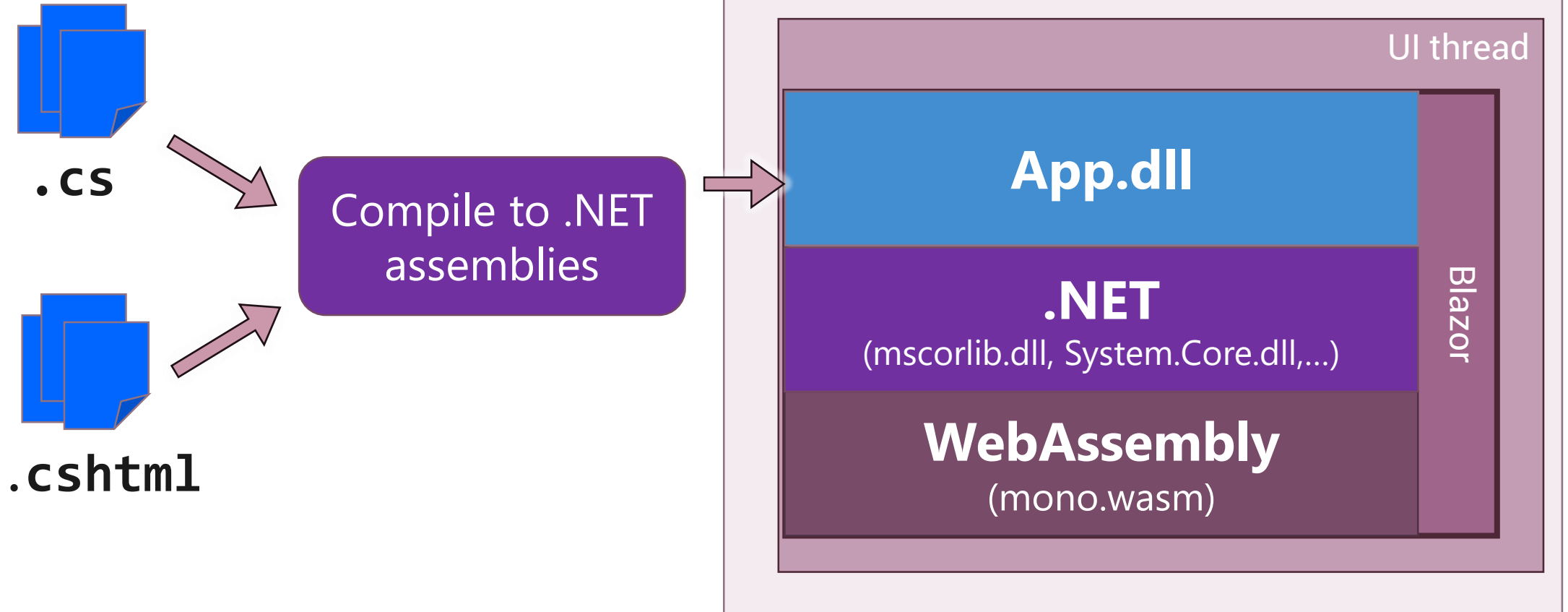
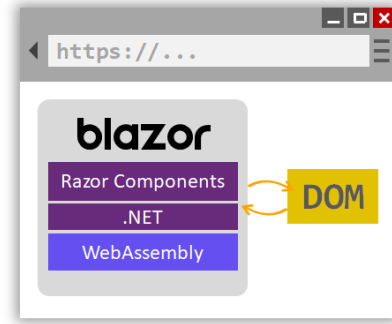
Blazor



This is in preparation for shipping Razor Components as a built-in feature of ASP.NET Core 3.0. *Note: client-side Blazor remains experimental while we continue to work on making the WebAssembly runtime complete.*

<https://github.com/aspnet/AspNetCore/issues/8931>

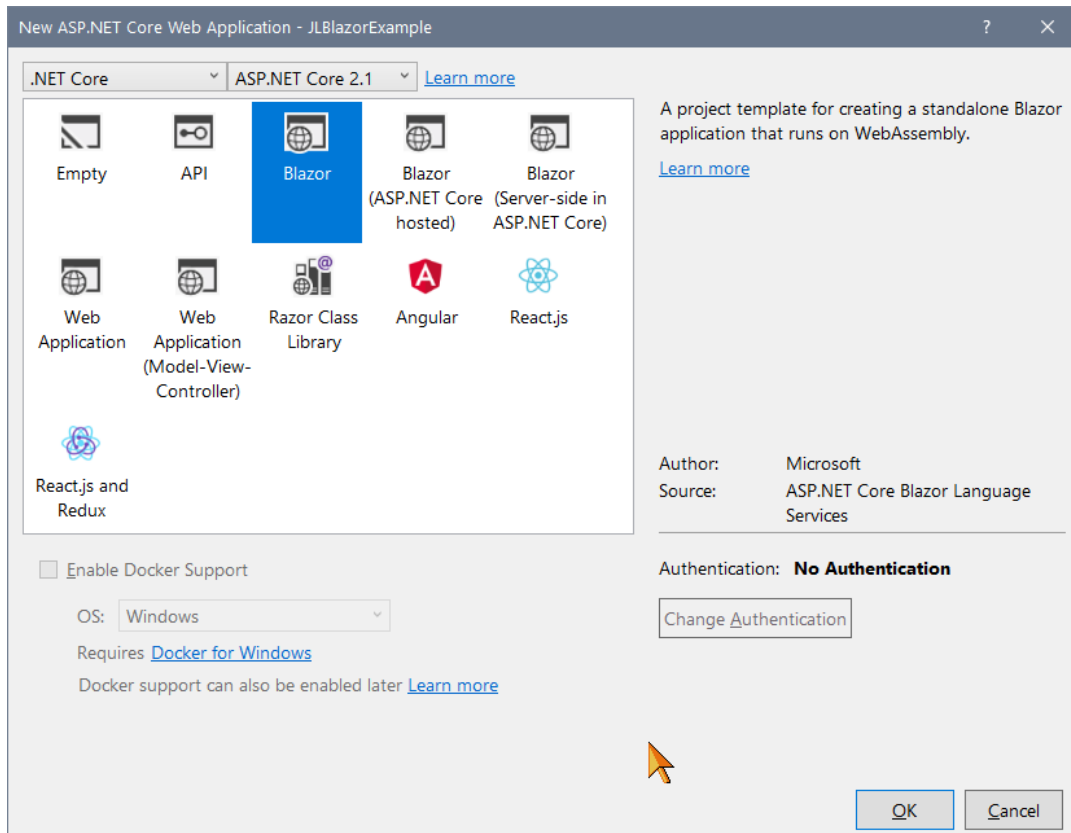
Blazor client-side



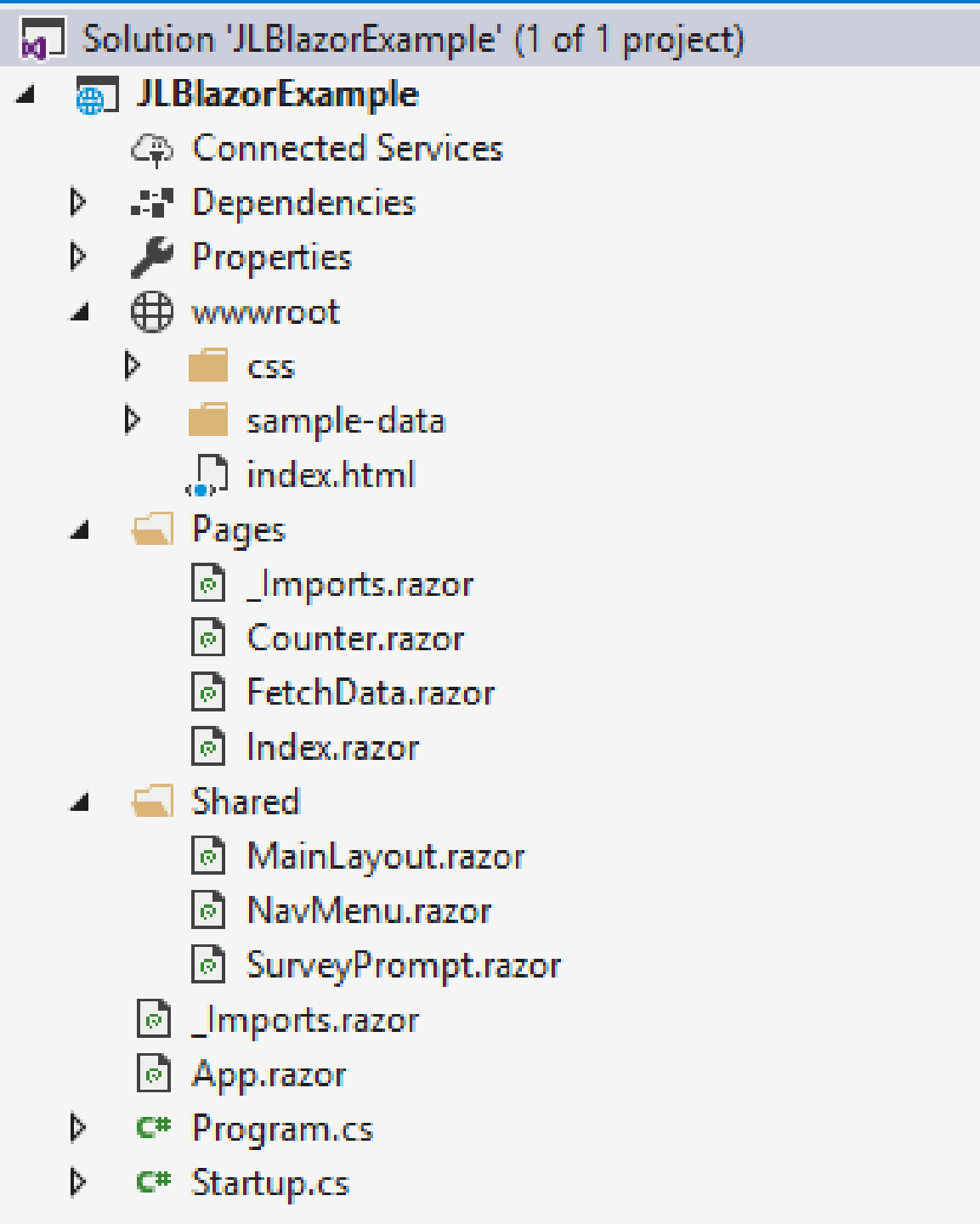
Blazor new project

Razor Component	razorcomponent	[C#]	Web/ASP.NET
Razor Page	page	[C#]	Web/ASP.NET
MVC ViewImports	viewimports	[C#]	Web/ASP.NET
MVC ViewStart	viewstart	[C#]	Web/ASP.NET
Blazor (server-side)	blazorserverside	[C#]	Web/Blazor
Blazor (ASP.NET Core hosted)	blazorhosted	[C#]	Web/Blazor/Hosted
Blazor Library	blazorlib	[C#]	Web/Blazor/Library
Blazor (client-side)	blazor	[C#]	Web/Blazor/Standalone
ASP.NET Core Empty	web	[C#], F#	Web/Empty
ASP.NET Core Web API	webapi	[C#], F#	Web/WebAPI
Razor Class Library	razorclasslib	[C#]	Web/Razor/Library/Razor Class Library

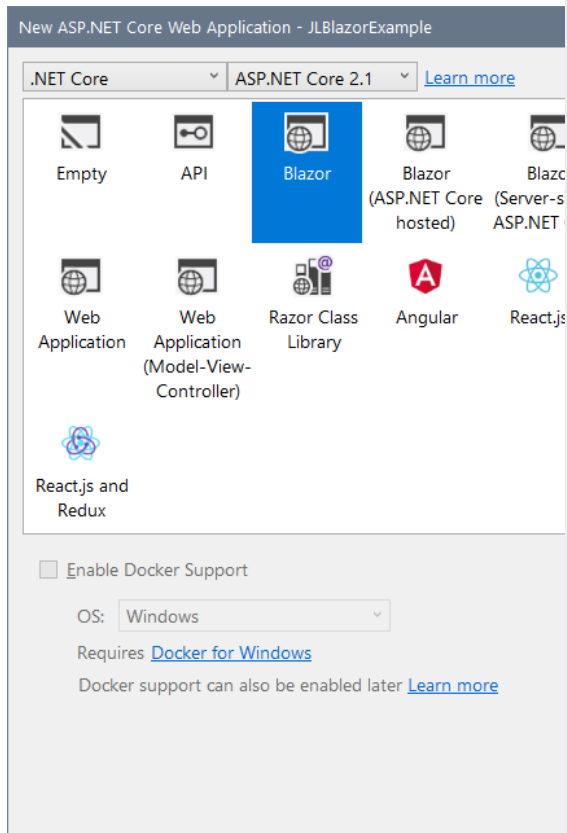
Blazor client-side



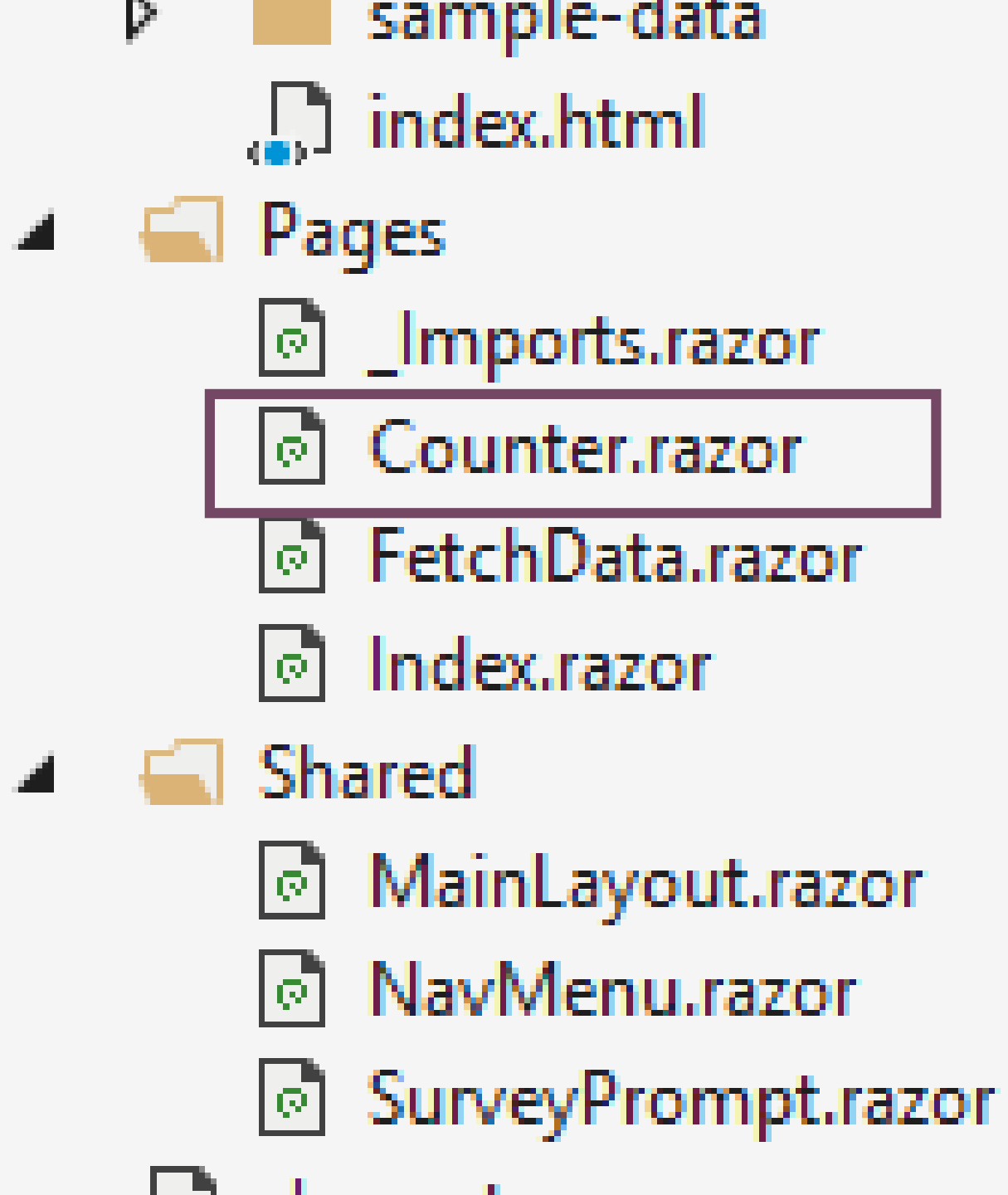
```
C:\Projects> dotnet new blazor -n  
JLBazorExample
```



Blazor client



C:\Projects> dotnet



JLBlazorExample

 Home Counter Fetch data

Hello, world!

Welcome to your new app.



How is Blazor working for you? Please take our [brief survey](#) and

Blazor: Sample

```
@page "/counter"  
@inject HttpClient
```

```
<h1>Counter</h1>  
<input type="number" bind="IncrementAmount" />  
<p>Current count: @currentCount</p>  
<button class="btn btn-primary" onclick="@IncrementCount">Click me</button>
```

```
@functions {  
    int currentCount = 0;  
    int IncrementAmount { get; set; } = 1;  
  
    void IncrementCount()  
    {  
        currentCount += IncrementAmount;  
    }  
}
```

Blazor: Routing

```
@page "/counter"
```

```
@inject HttpClient
```

```
<h1>Counter</h1>
```

```
<input type="number" bind="IncrementAmount" />
```

```
<p>Current count: @currentCount</p>
```

```
<button class="btn btn-primary" onclick="@IncrementCount">Click me</button>
```

```
@functions {
```

```
    int currentCount = 0;
```

```
    int IncrementAmount { get; set; } = 1;
```

```
    void IncrementCount()
```

```
{
```

```
        currentCount += IncrementAmount;
```

```
}
```

```
}
```


Blazor: Razor template

```
@page "/counter"  
@inject HttpClient
```

```
<h1>Counter</h1>  
<input type="number" bind="IncrementAmount" />  
<p>Current count: @currentCount</p>  
<button class="btn btn-primary" onclick="@IncrementCount">Click me</button>
```

```
@functions {  
    int currentCount = 0;  
    int IncrementAmount { get; set; } = 1;  
  
    void IncrementCount()  
    {  
        currentCount += IncrementAmount;  
    }  
}
```

Blazor: C#

```
@page "/counter"  
@inject HttpClient
```

```
<h1>Counter</h1>  
<input type="number" bind="IncrementAmount" />  
<p>Current count: @currentCount</p>  
<button class="btn btn-primary" onclick="@IncrementCount">Click me</button>
```

```
@functions {  
    int currentCount = 0;  
    int IncrementAmount { get; set; } = 1;  
  
    void IncrementCount()  
    {  
        currentCount += IncrementAmount;  
    }  
}
```

Blazor: One way data binding

```
@page "/counter"  
@inject HttpClient
```

```
<h1>Counter</h1>  
<input type="number" bind="IncrementAmount" />  
<p>Current count: @currentCount</p>  
<button class="btn btn-primary" onclick="@IncrementCount">Click me</button>
```

```
@functions {  
    int currentCount = 0;  
    int IncrementAmount { get; set; } = 1;  
  
    void IncrementCount()  
    {  
        currentCount += IncrementAmount;  
    }  
}
```

Blazor: Two way data binding

```
@page "/counter"  
@inject HttpClient
```

```
<h1>Counter</h1>  
<input type="number" bind="IncrementAmount" />  
<p>Current count: @currentCount</p>  
<button class="btn btn-primary" onclick="@IncrementCount">Click me</button>
```

```
@functions {  
    int currentCount = 0;  
    int IncrementAmount { get; set; } = 1;  
  
    void IncrementCount()  
    {  
        currentCount += IncrementAmount;  
    }  
}
```

Blazor: Event binding

```
@page "/counter"  
@inject HttpClient
```

```
<h1>Counter</h1>  
<input type="number" bind="IncrementAmount" />  
<p>Current count: @currentCount</p>  
<button class="btn btn-primary" onclick="@IncrementCount">Click me</button>
```

```
@functions {  
    int currentCount = 0;  
    int IncrementAmount { get; set; } = 1;
```

```
    void IncrementCount()  
    {  
        currentCount += IncrementAmount;  
    }
```

```
}
```

Blazor: Using

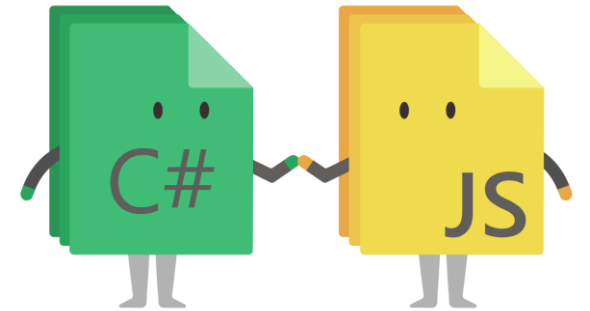
```
@page "/counter"
```

```
@inject HttpClient
```

```
<h1>Counter</h1>  
<input type="number" bind="IncrementAmount" />  
<p>Current count: @currentCount</p>  
<button class="btn btn-primary" onclick="@IncrementCount">Click me</button>
```

```
@functions {  
    int currentCount = 0;  
    int IncrementAmount { get; set; } = 1;  
  
    void IncrementCount()  
    {  
        currentCount += IncrementAmount;  
    }  
}
```

JavaScript interop



```
<button onclick="@CallDoSomething">Do something</button>
```

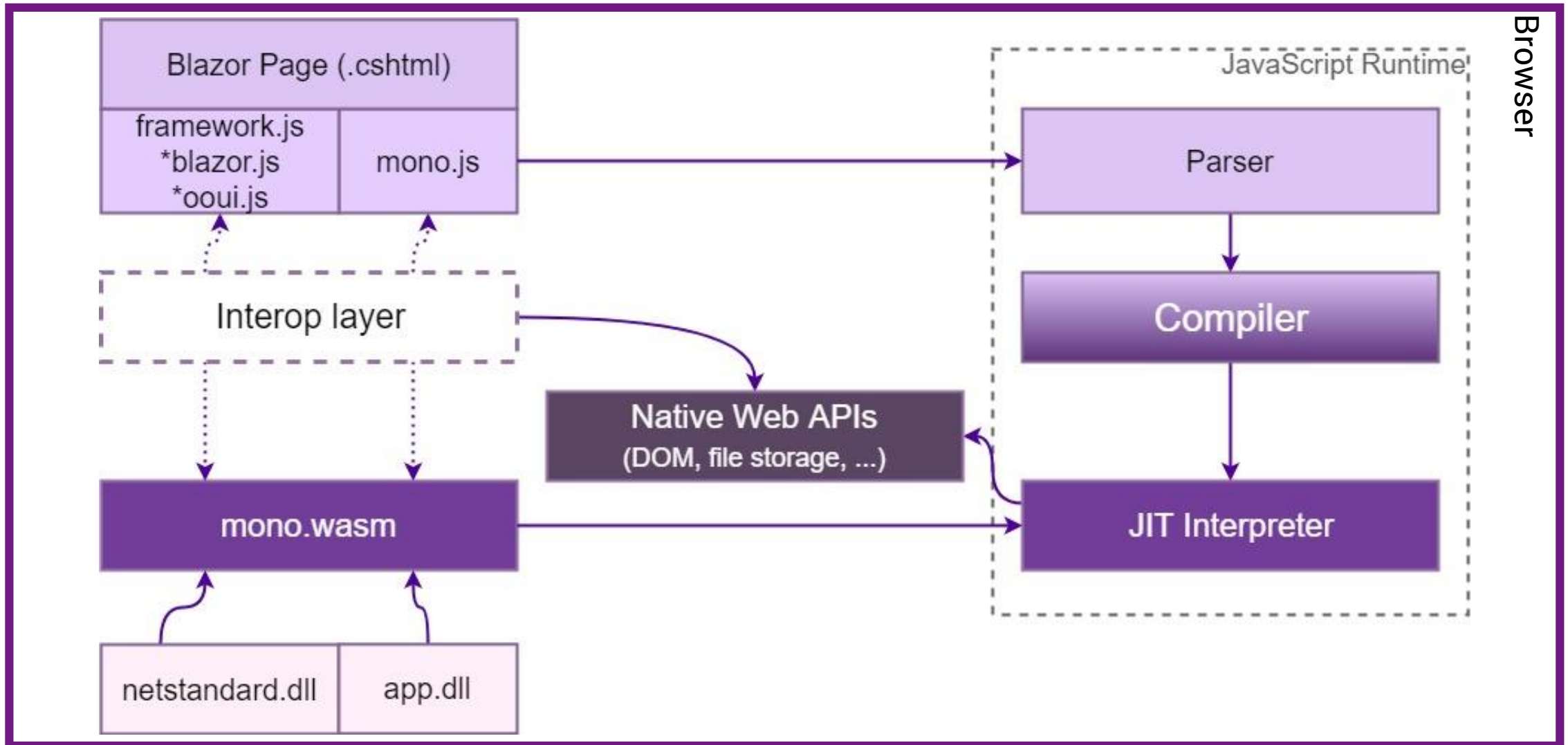
```
<script>
  function doSomething(message) {
    console.log(message);
    return true;
  }

```

```
Blazor.registerFunction('doSomething', doSomething);
</script>
```

```
@functions {
  public void CallDoSomething()
  {
    RegisteredFunction.Invoke<bool>("doSomething", "Hello World");
  }
}
```

WASM + C#



Blazor sample in browser

The screenshot shows a web browser window with the address bar set to `localhost:64814`. The page title is `JLBlazorExample`. The main content area displays "Hello, world!" and "Welcome to your new app." A sidebar on the left contains links for "Home" and "Counter". The browser's developer tools are open, showing the Network tab. The table below lists the requests made by the browser.

Stan	Metoda	Domena	Plik	Przycz...	Typ	Przesłano	Rozmiar	0 ms	320 ms	640 ms	960 ms	1,28 s	1,6 s
304	GET	localhost:64814	/	document	html	0	426 B	5 ms					
304	GET	localhost:64814	bootstrap.min.css	stylesheet	css	0	141,49 KB	6 ms					
304	GET	localhost:64814	site.css	stylesheet	css	0	2,20 KB	7 ms					
304	GET	localhost:64814	blazor.webassembly.js	script	js	0	0 B	6 ms					
304	GET	localhost:64814	open-iconic-bootstrap.min.css	stylesheet	css	0	9,17 KB	4 ms					
304	GET	localhost:64814	blazor.boot.json	fetch	json	0	590 B	3 ms					
200	GET	localhost:64814	favicon.ico	img	html	0	426 B						
304	GET	localhost:64814	mono.js	script	js	0	0 B	3 ms					
304	GET	localhost:64814	mono.wasm	fetch	wasm	0	1,83 MB	3 ms					
200	GET	localhost:64814	JLBlazorExample.dll	xhr	octet-stream	7,19 KB	16 KB				6 ms		
304	GET	localhost:64814	JL.GameOfLife.Core.dll	xhr	octet-stream	0	12 KB				6 ms		
200	GET	localhost:64814	Microsoft.AspNetCore.Blazor.Browser.dll	xhr	octet-stream	14,76 KB	28,50 KB				7 ms		
200	GET	localhost:64814	Microsoft.AspNetCore.Blazor.dll	xhr	octet-stream	40,32 KB	91 KB				10 ms		
200	GET	localhost:64814	Microsoft.AspNetCore.Blazor.TagHelper...	xhr	octet-stream	2,42 KB	5 KB				4 ms		
200	GET	localhost:64814	Microsoft.Extensions.DependencyInjecti...	xhr	octet-stream	11,89 KB	26,50 KB				7 ms		
200	GET	localhost:64814	Microsoft.Extensions.DependencyInjecti...	xhr	octet-stream	20,44 KB	42 KB				7 ms		
200	GET	localhost:64814	System.Net.Http.dll	xhr	octet-stream	31,82 KB	65,50 KB				8 ms		
304	GET	localhost:64814	JL.GameOfLife.Core.pdb	xhr	octet-stream	0	3,34 KB				4 ms		
200	GET	localhost:64814	JLBlazorExample.pdb	xhr	octet-stream	1,87 KB	2,36 KB						
200	GET	localhost:64814	Microsoft.JSInterop.dll	xhr	octet-stream	20,67 KB	40,50 KB						
200	GET	localhost:64814	Mono.WebAssembly.Interop.dll	xhr	octet-stream	3,15 KB	6 KB						
304	GET	localhost:64814	mscorlib.dll	xhr	octet-stream	0	1,56 MB						
200	GET	localhost:64814	System.Core.dll	xhr	octet-stream	137,22 KB	334 KB						

24 zapytania | Przesłano: 4,29 MB / 291,75 KB | 1,31 s | DOMContentLoaded: 239 ms | load: 244 ms



 bootstrap.min.css	stylesheet	css	0	141,49 KB
 site.css	stylesheet	css	0	2,20 KB
 blazor.webassembly.js	script	js	0	0 B
 open-iconic-bootstrap.min.css	stylesheet	css	0	9,17 KB
 blazor.boot.json	fetch	json	0	590 B
 favicon.ico	img	html	0	426 B
 mono.js	script	js	0	0 B
 mono.wasm	fetch	wasm	0	1,83 MB
 JLBazorExample.dll	xhr	octet-stream	7,19 KB	16 KB
 JL.GameOfLife.Core.dll	xhr	octet-stream	0	12 KB
 Microsoft.AspNetCore.Blazor.Browser.dll	xhr	octet-stream	14,76 KB	28,50 KB
 Microsoft.AspNetCore.Blazor.dll	xhr	octet-stream	40,32 KB	91 KB
 Microsoft.AspNetCore.Blazor.TagHelper...	xhr	octet-stream	2,42 KB	5 KB

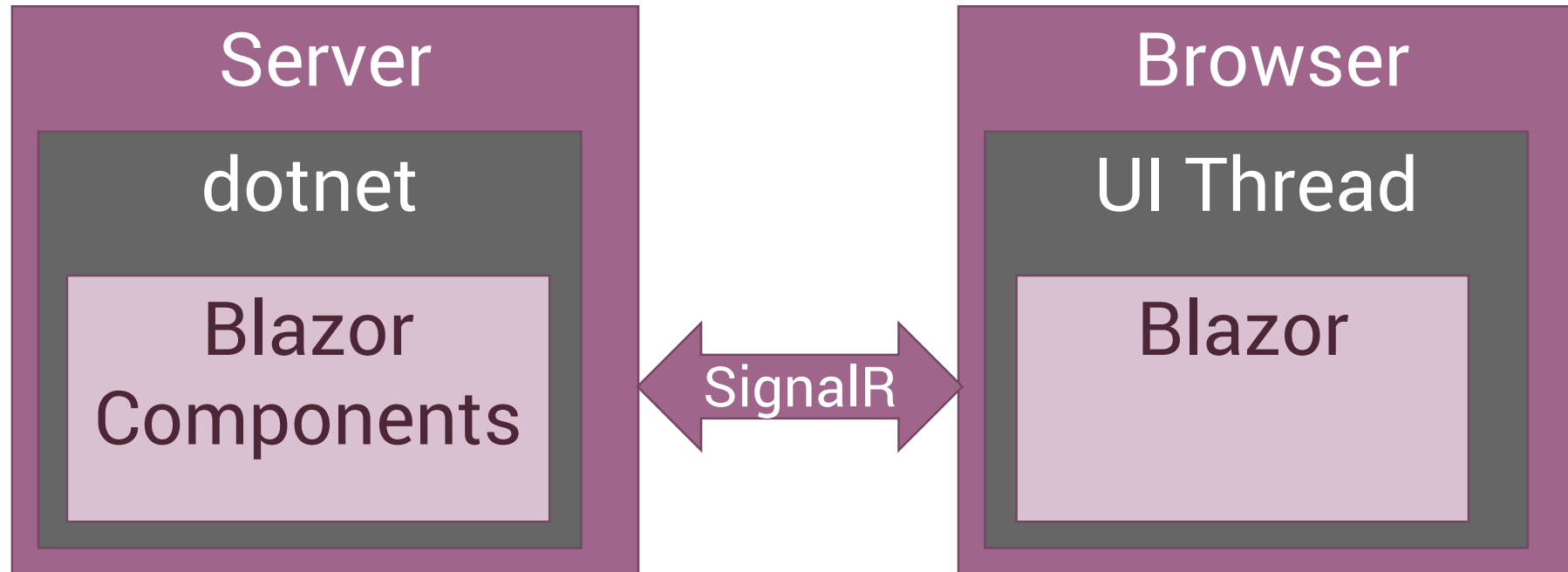
 bootstrap.min.css	stylesheet	css	0	141,49 KB
 site.css	stylesheet	css	0	2,20 KB
 blazor.webassembly.js	script	js	0	0 B
 open-iconic-bootstrap.min.css	stylesheet	css	0	9,17 KB
 blazor.boot.json	fetch	json	0	590 B
 favicon.ico	img	html	0	426 B
 mono.js	script	js	0	0 B
 mono.wasm	fetch	wasm	0	1,83 MB
 JLBazorExample.dll	xhr	octet-stream	7,19 KB	16 KB
 JL.GameOfLife.Core.dll	xhr	octet-stream	0	12 KB
 Microsoft.AspNetCore.Blazor.Browser.dll	xhr	octet-stream	14,76 KB	28,50 KB
 Microsoft.AspNetCore.Blazor.dll	xhr	octet-stream	40,32 KB	91 KB
 Microsoft.AspNetCore.Blazor.TagHelper...	xhr	octet-stream	2,42 KB	5 KB

 bootstrap.min.css	stylesheet	css	0	141,49 KB
 site.css	stylesheet	css	0	2,20 KB
 blazor.webassembly.js	script	js	0	0 B
 open-iconic-bootstrap.min.css	stylesheet	css	0	9,17 KB
 blazor.boot.json	fetch	json	0	590 B
 favicon.ico	img	html	0	426 B
 mono.js	script	js	0	0 B
 mono.wasm	fetch	wasm	0	1,83 MB
 JLBazorExample.dll	xhr	octet-stream	7,19 KB	16 KB
 JL.GameOfLife.Core.dll	xhr	octet-stream	0	12 KB
 Microsoft.AspNetCore.Blazor.Browser.dll	xhr	octet-stream	14,76 KB	28,50 KB
 Microsoft.AspNetCore.Blazor.dll	xhr	octet-stream	40,32 KB	91 KB
 Microsoft.AspNetCore.Blazor.TagHelper...	xhr	octet-stream	2,42 KB	5 KB

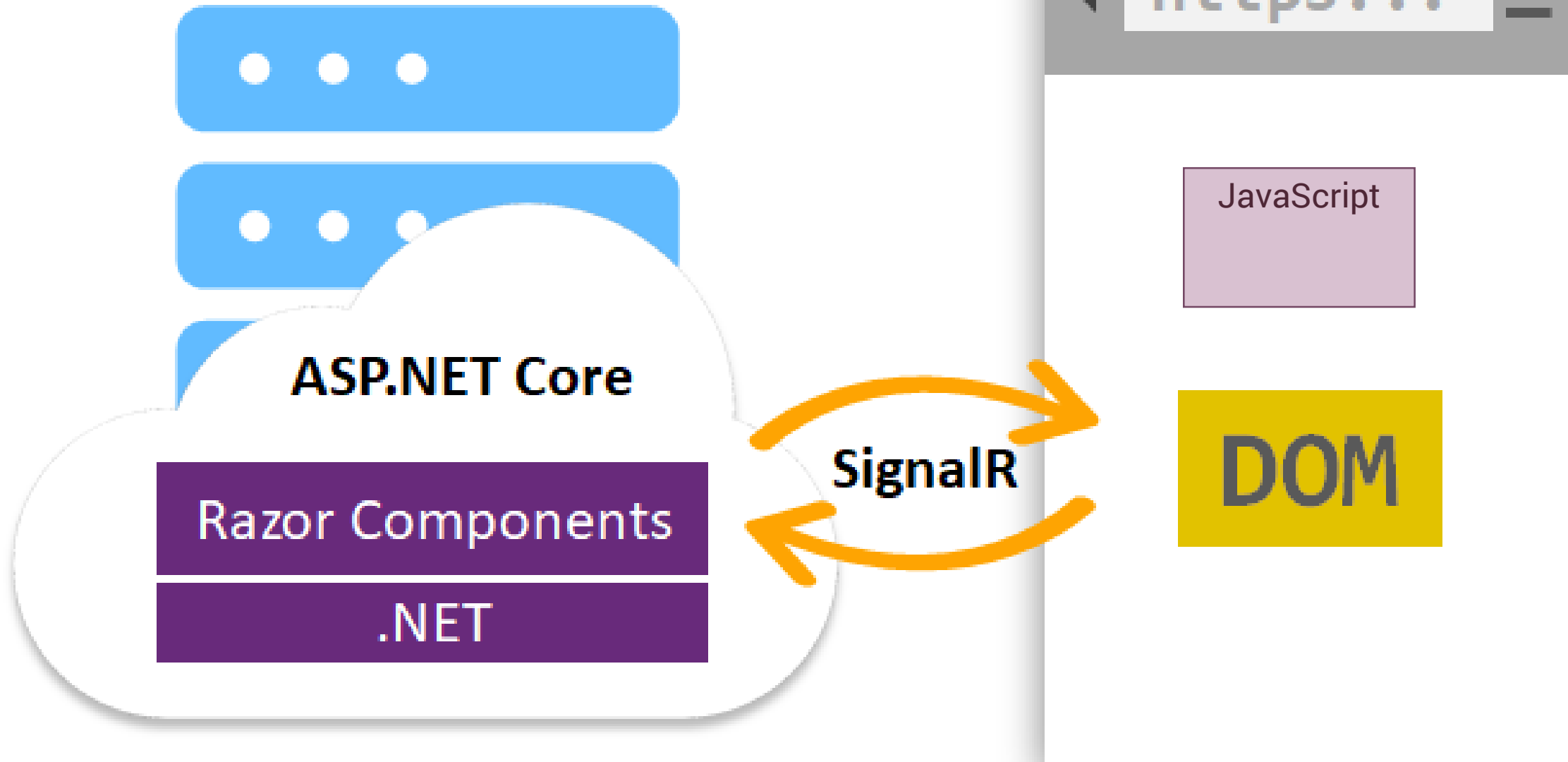
 bootstrap.min.css	stylesheet	css	0	141,49 KB
 site.css	stylesheet	css	0	2,20 KB
 blazor.webassembly.js	script	js	0	0 B
 open-iconic-bootstrap.min.css	stylesheet	css	0	9,17 KB
 blazor.boot.json	fetch	json	0	590 B
 favicon.ico	img	html	0	426 B
 mono.js	script	js	0	0 B
 mono.wasm	fetch	wasm	0	1,83 MB
 JLBazorExample.dll	xhr	octet-stream	7,19 KB	16 KB
 JL.GameOfLife.Core.dll	xhr	octet-stream	0	12 KB
 Microsoft.AspNetCore.Blazor.Browser.dll	xhr	octet-stream	14,76 KB	28,50 KB
 Microsoft.AspNetCore.Blazor.dll	xhr	octet-stream	40,32 KB	91 KB
 Microsoft.AspNetCore.Blazor.TagHelper...	xhr	octet-stream	2,42 KB	5 KB

 bootstrap.min.css	stylesheet	css	0	141,49 KB
 site.css	stylesheet	css	0	2,20 KB
 blazor.webassembly.js	script	js	0	0 B
 open-iconic-bootstrap.min.css	stylesheet	css	0	9,17 KB
 blazor.boot.json	fetch	json	0	590 B
 favicon.ico	img	html	0	426 B
 mono.js	script	js	0	0 B
 mono.wasm	fetch	wasm	0	1,83 MB
 JLBazorExample.dll	xhr	octet-stream	7,19 KB	16 KB
 JL.GameOfLife.Core.dll	xhr	octet-stream	0	12 KB
 Microsoft.AspNetCore.Blazor.Browser.dll	xhr	octet-stream	14,76 KB	28,50 KB
 Microsoft.AspNetCore.Blazor.dll	xhr	octet-stream	40,32 KB	91 KB
 Microsoft.AspNetCore.Blazor.TagHelper...	xhr	octet-stream	2,42 KB	5 KB

Let's switch elements places



Blazor server-side

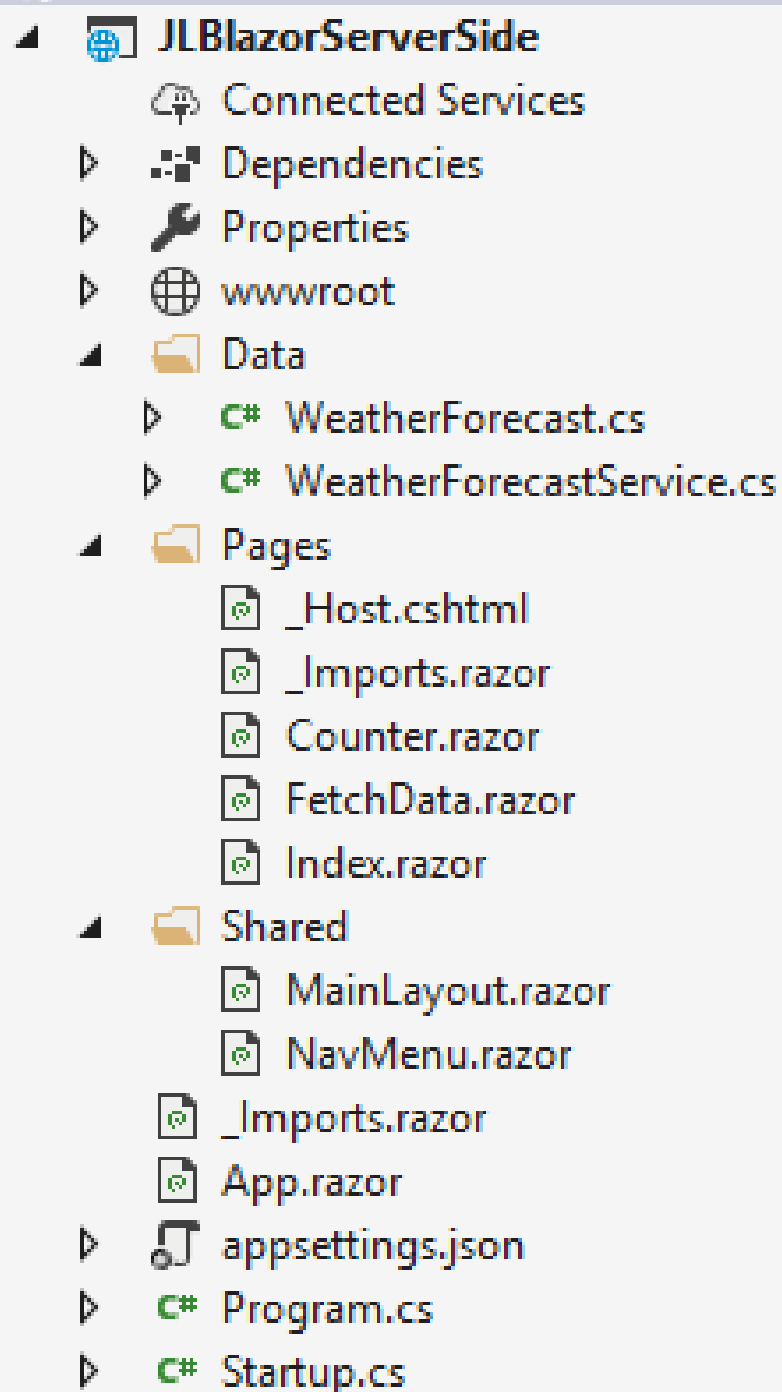


Blazor server-side

```
C:\Projects> dotnet new  
blazorserverside -n  
JLBlazorServerSide
```

Blazor server-side

```
C:\Projects> dotnet  
blazorserver  
JLBlazorServerSide
```



Solution 'JLBlazorExample' (1 of 1 project)

JLBlazorExample

- Connected Services
- Dependencies
- Properties
- wwwroot
 - css
 - sample-data
 - index.html
- Pages
 - _Imports.razor
 - Counter.razor
 - FetchData.razor
 - Index.razor
- Shared
 - MainLayout.razor
 - NavMenu.razor
 - SurveyPrompt.razor
 - _Imports.razor
 - App.razor
- Program.cs
- Startup.cs

JLBlazorServerSide

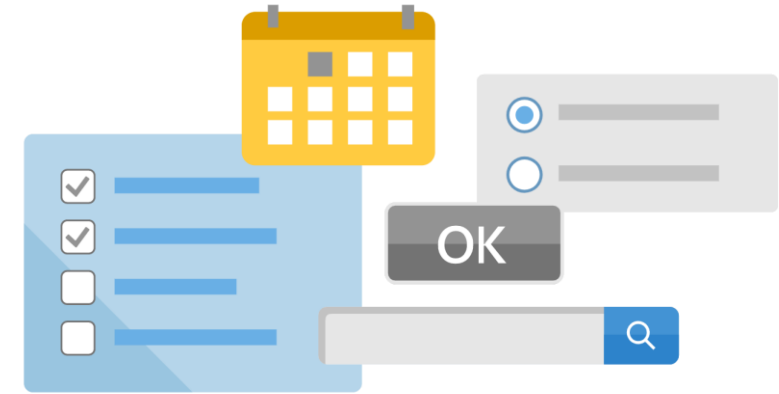
- Connected Services
- Dependencies
- Properties
- wwwroot
- Data
 - WeatherForecast.cs
 - WeatherForecastService.cs
- Pages
 - _Host.cshtml
 - _Imports.razor
 - Counter.razor
 - FetchData.razor
 - Index.razor
- Shared
 - MainLayout.razor
 - NavMenu.razor
 - _Imports.razor
 - App.razor
- appsettings.json
- Program.cs
- Startup.cs

```
public class Program { //Blazor client side
    public static IWebAssemblyHostBuilder CreateHostBuilder(string[] args) =>
        BlazorWebAssemblyHost.CreateDefaultBuilder().UseBlazorStartup<Startup>();
}
```

```
public class Program { //Blazor server side
    public static IHostBuilder CreateHostBuilder(string[] args) =>
        Host.CreateDefaultBuilder(args).ConfigureWebHostDefaults(webBuilder =>
            {webBuilder.UseStartup<Startup>();});
}
```

```
public class Startup {
    public void ConfigureServices(IServiceCollection services) {
        services.AddRazorPages();
        services.AddServerSideBlazor();
        services.AddSingleton<WeatherForecastService>();
    }
}
```

UI component ecosystem



+ Add Product			
Product Name	Unit Price ↑	Units In Stock	
Product30	\$94.20	30	Edit Delete
Product31	\$97.34	31	Edit Delete
Product32	\$100.48	32	Edit Delete
Product33	103.62	36	Update Cancel
Product34	\$106.76	34	Edit Delete
Product35	\$109.90	35	Edit Delete
Product36	\$113.04	36	Edit Delete
Product37	\$116.18	37	Edit Delete

◀ ◁ 1 2 3 4 5 6 7 ▶ ▷ 31 - 45 of 100

Telerik UI for Blazor

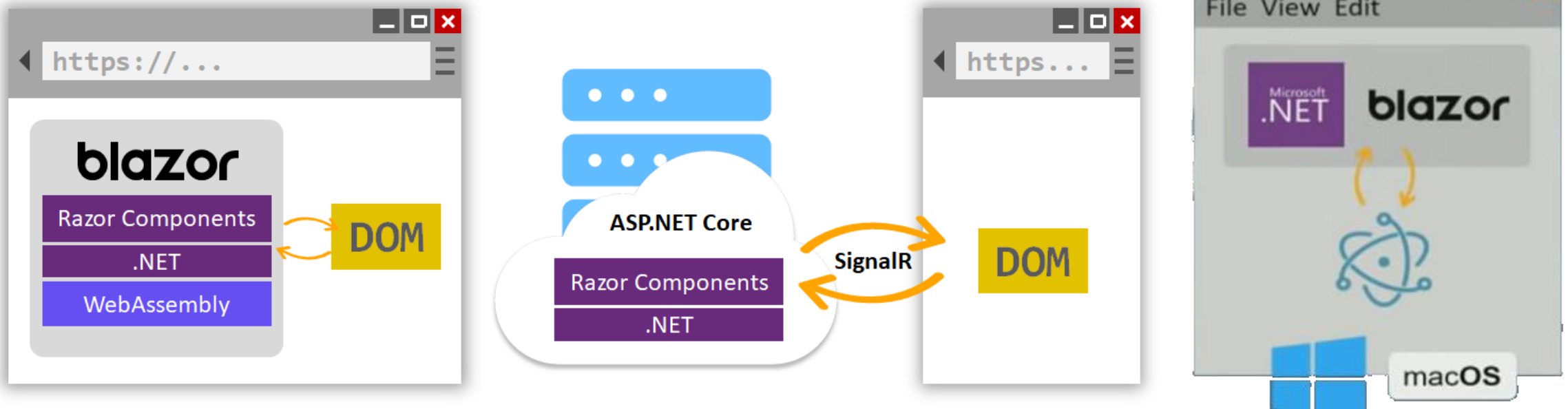
from **\$899**
/developer, perpetual license

 **DevExpress**



 **Syncfusion**

Blazor: possibilities



<https://www.youtube.com/watch?v=qrf3OrLHeFI>

Steve Sanderson - Blazor, a New .NET Single Page Application Framework | Øredev 2018

What's the future?



What WebAssembly are next to?



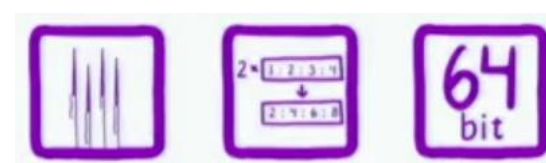
Loadtime improvements

- streaming compilation ●
- tiered compilation ●
- implicit HTTP caching ●
- other improvements



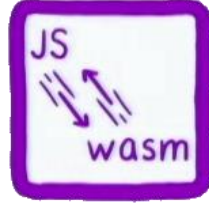
Making use of modern Hardware

- threading ●
- SIMD
- wasm 64bit

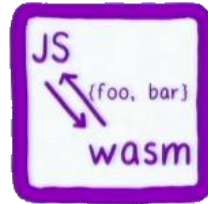


What WebAssembly are next to?

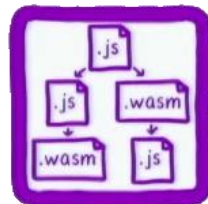
- fast calls
JS $\leftrightarrow$ wasm



- esy and fast
data exchange



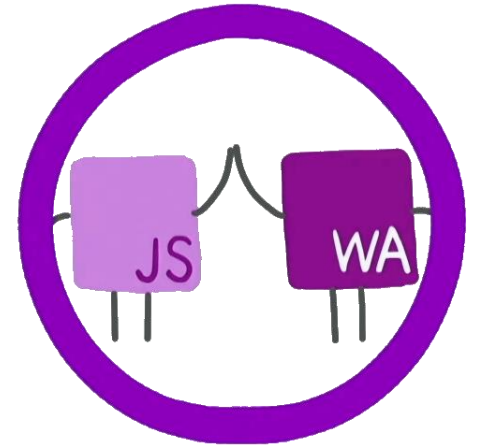
- ES module
integration



- toolchain integration
like npm or webpack



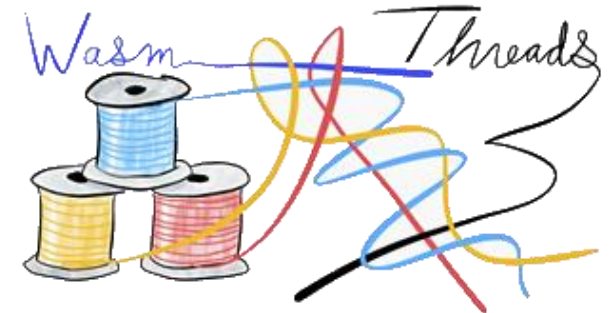
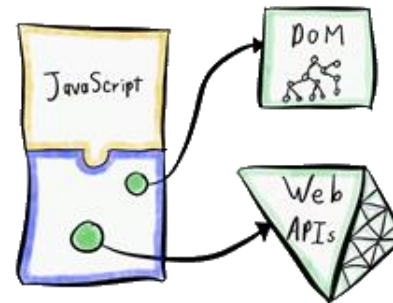
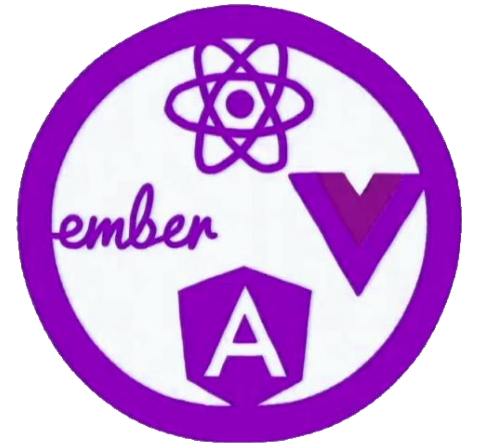
- backward compatibility



What WebAssembly are next to?

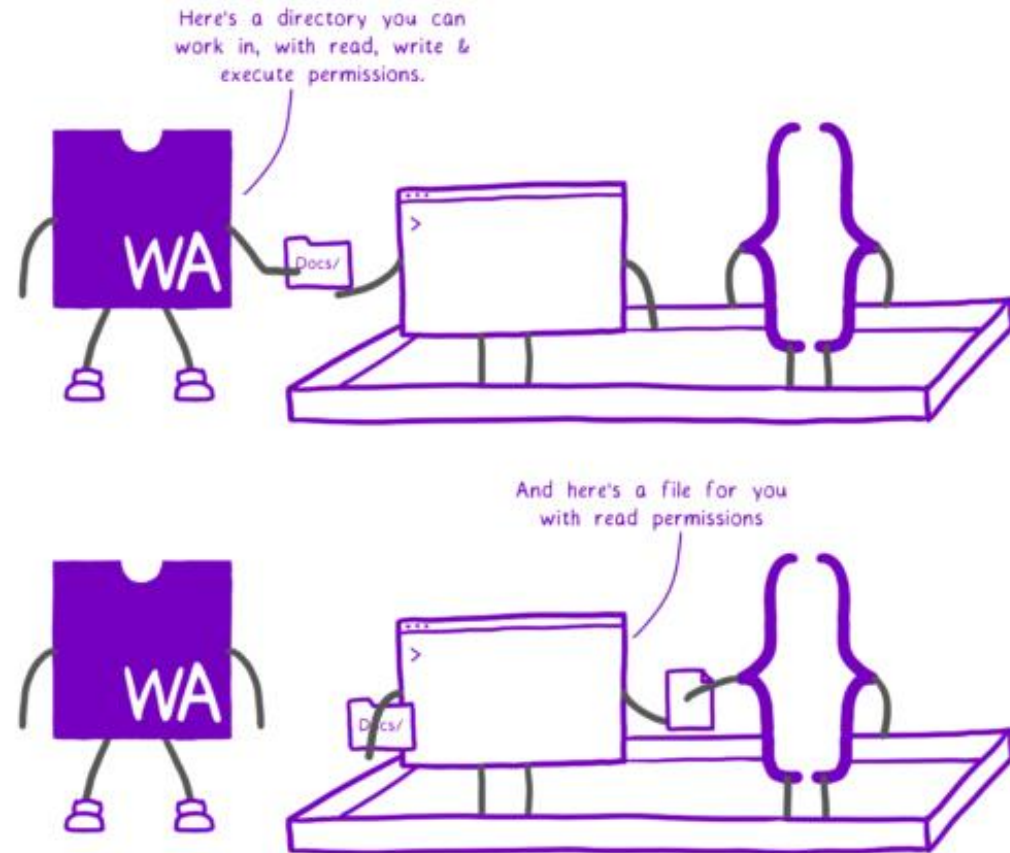
High level language features

- GC integration
- Exception handling
- Debugging
- Tail calls

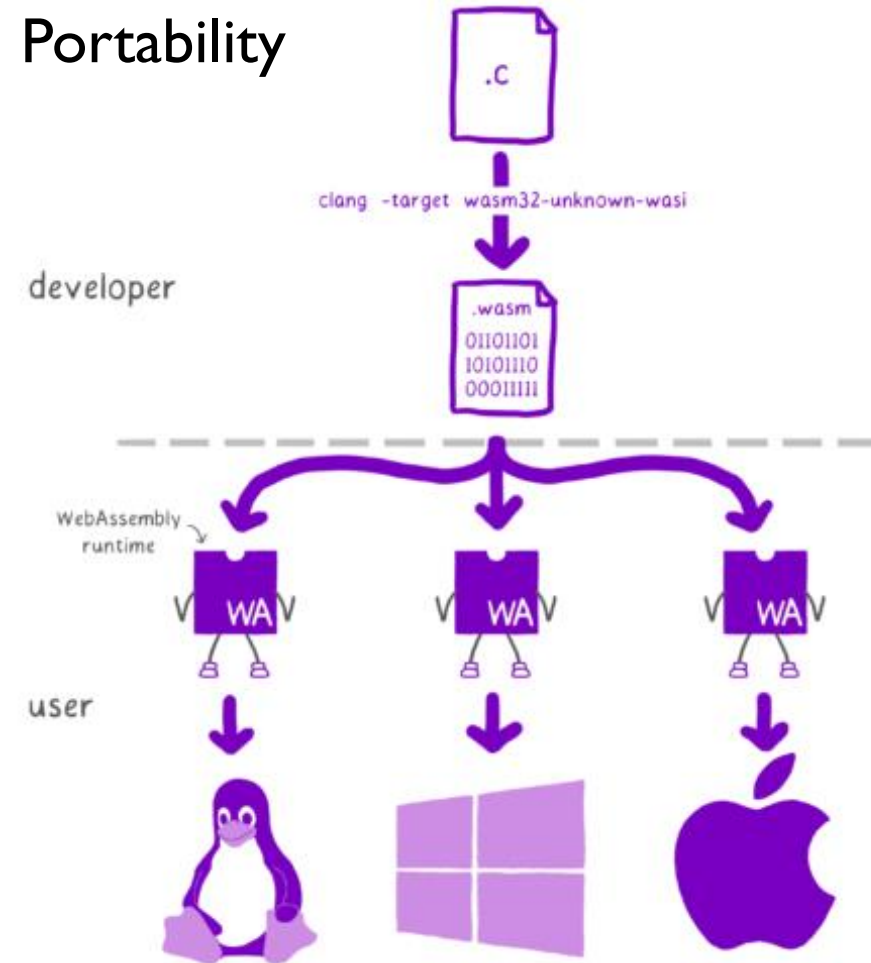


WASI – WebAssembly System Interface

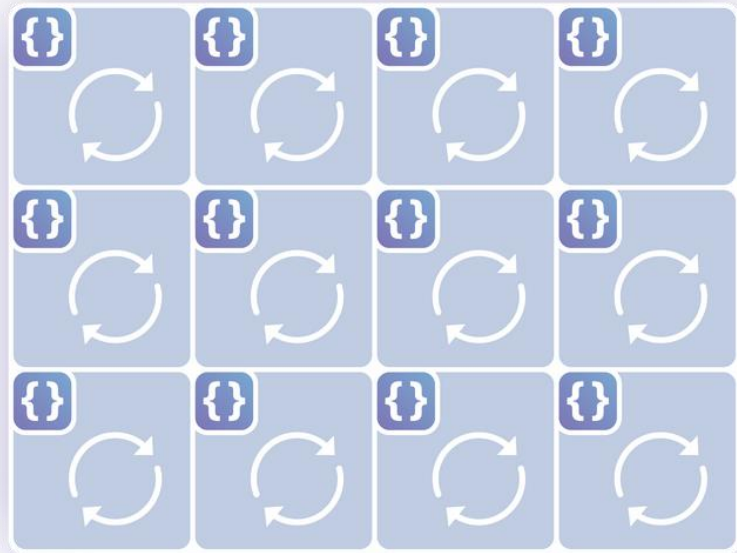
Security



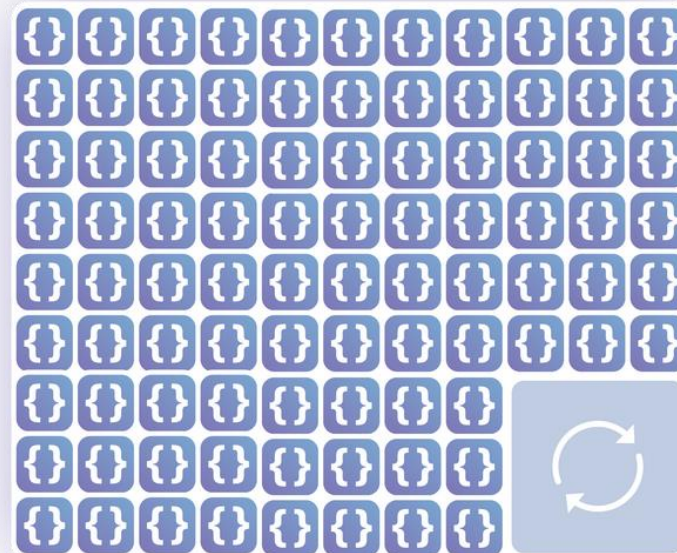
Portability



CloudFlare



Virtual machine



Isolate model



User code



Process overhead



Context switching



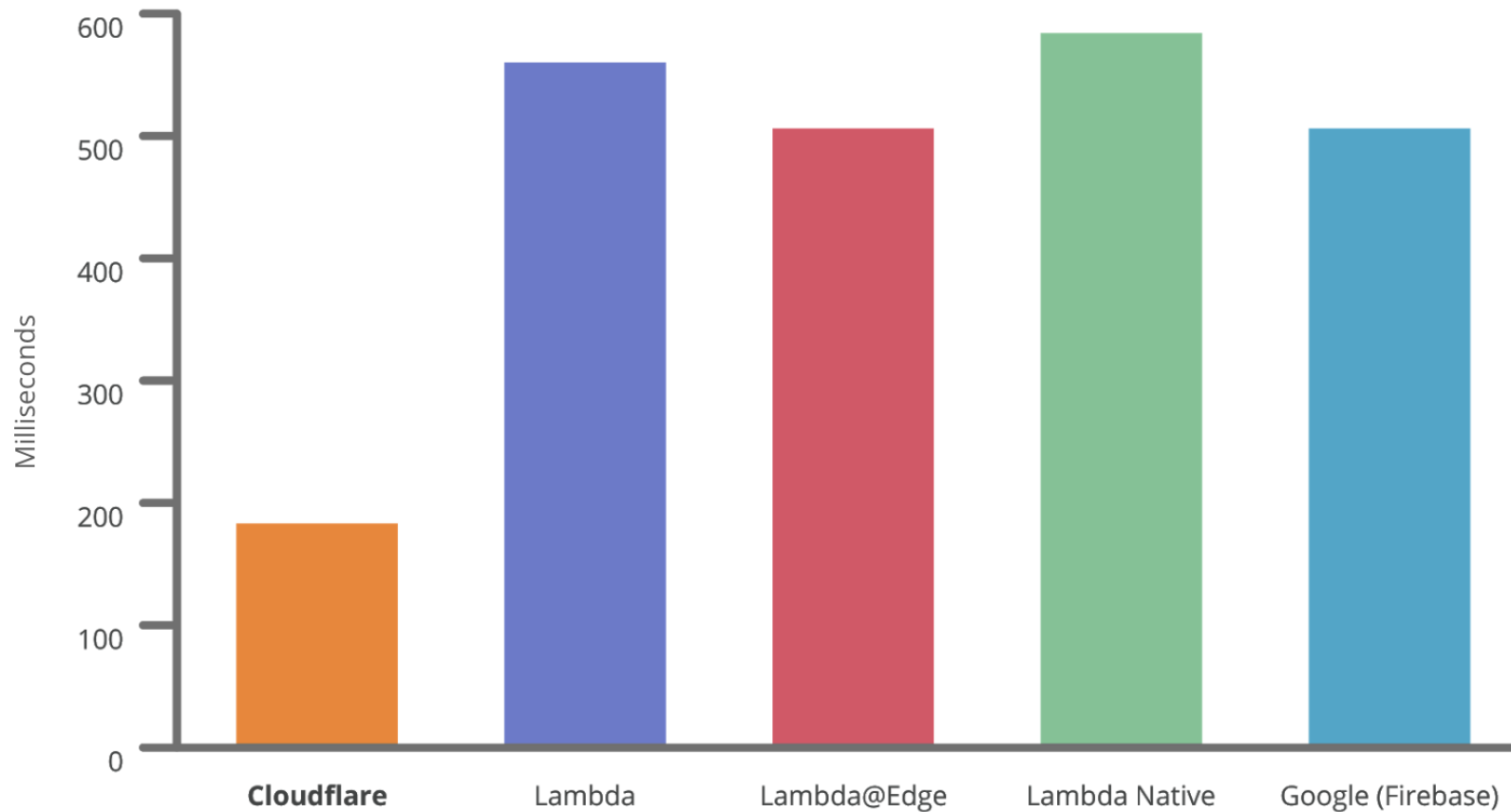
Multitasking

Time

<https://blog.cloudflare.com/cloud-computing-without-containers/>
<https://www.youtube.com/watch?v=A9SydP1CcZU>

CloudFlare no more cold starts?

Request response time for primary serverless providers



So WebAssembly?

Possibilities

Flexibility

Diversity

Cross-platform

Promise of bright tomorrow for today's projects

Questions??



THANKS

Joanna Lamch

JLamch@gmail.com

JLamch.net

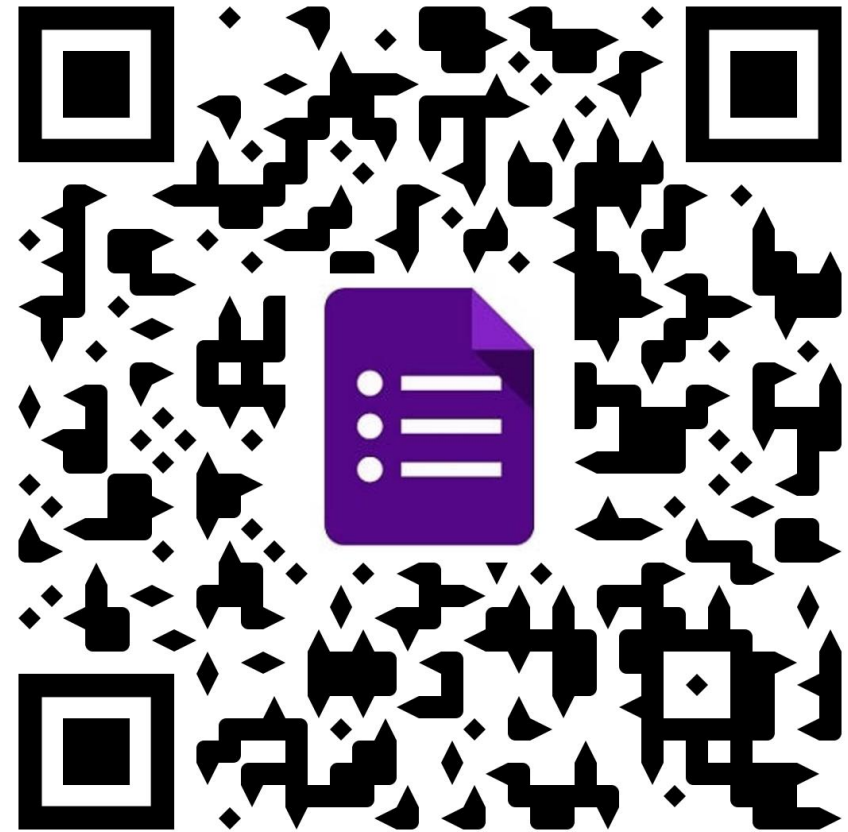
ProgramistkaKot.pl



- Ankieta

Działanie pociąga za sobą koszty i ryzyko, ale o wiele mniejsze niż te, które wiążą się z wygodną bezczynnością

- Będzie mi miło móc ulepszyć moją prezentację dzięki Twoim komentarzom, dlatego proszę Cię o wypełnienie ankiety, bądź kontakt mailowy.



• Bibliografia

- <https://webassembly.org/docs/high-level-goals/>
- <https://www.smashingmagazine.com/2017/05/abridged-cartoon-introduction-webassembly/>
- https://www.youtube.com/watch?v=HktWin_LPf4&feature=youtu.be
- <https://www.youtube.com/watch?v=pBYqen3B2gc>
- <https://www.youtube.com/watch?v=BnYq7JapeDA>
<https://www.youtube.com/watch?v=kS29TT4wk44&feature=youtu.be>
- <https://github.com/mbasso/awesome-wasm>
- <https://github.com/migueldeicaza/mono-wasm>
- <https://superkotlin.com/kotlin-and-webassembly/>
- <https://medium.com/@mumarov/how-to-get-started-with-kotlin-native-and-web-assembly-baa2813f0d9>
- <https://github.com/DenisKolodin/yew>
- <https://www.mergeconflict.fm/89>
- <https://dotnetrocks.com/?show=1539>
- <https://dotnetrocks.com/?show=1540>
- <https://dotnetrocks.com/?show=1537>
- <https://www.hanselman.com/blog/NETAndWebAssemblyIsThisTheFutureOfTheFrontend.aspx>
<https://hacks.mozilla.org/2018/04/sneak-peek-at-webassembly-studio/>
- https://github.com/migueldeicaza/mono-wasm?WT.mc_id=blog-scottha
- <https://blog.scottlogic.com/ceberhardt/>
- <https://blog.logrocket.com/working-with-the-blazor-javascript-interop-3c2a8d0eb56c>
<https://s3.amazonaws.com/mozilla-games/ZenGarden/EpicZenGarden.html>
<https://blog.logrocket.com/working-with-the-blazor-javascript-interop-3c2a8d0eb56c>

