DEV CAFE
Connect and share

Blazor WebAssembly

Perfect for SPA and much more

Thanh Le

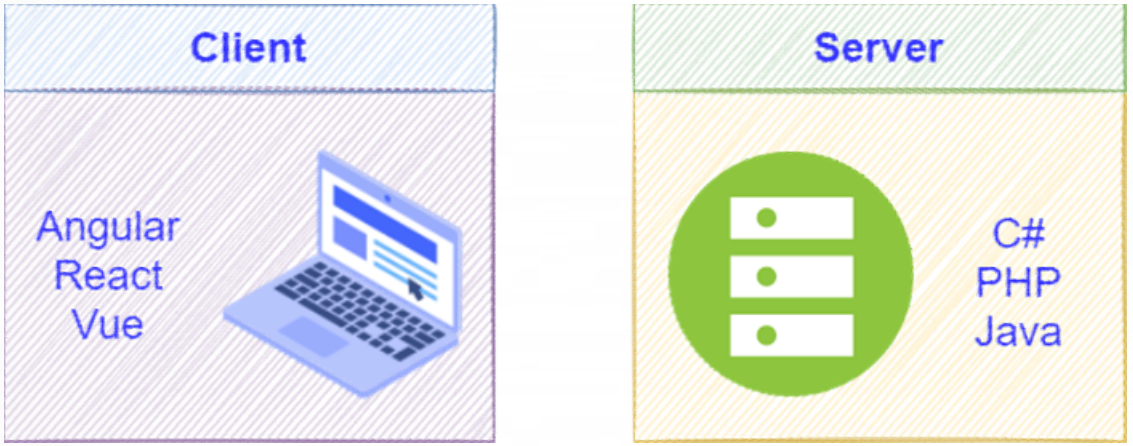
DEV CAFE
Connect and share

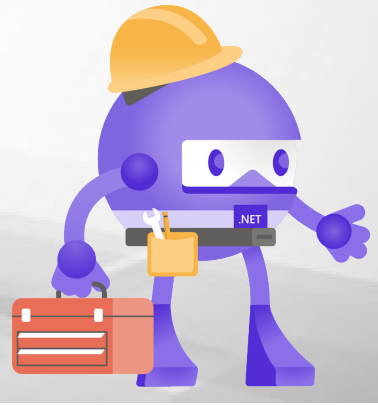
Agenda

1. What is WebAssembly?
2. Blazor Overview
3. Blazor WebAssembly vs Blazor Server
4. PROS & CONS of Blazor hosting models
5. Demo

DEV CAFE
Connect and share

Modern Web Development





What is WebAssembly?

DEV CAFE
Connect and share

DEV CAFE
Connect and share

Web Assembly

"WebAssembly (abbreviated Wasm) is a binary instruction format for a stack-based virtual machine. Wasm is designed as a portable target for compilation of high-level languages like C/C++/Rust, enabling deployment on the web for client and server applications."

Source: https://webassembly.org/

WA

- **Recommended by W3C (5th Dec 2019)**
- **Can written by C/C++, C#, Python, Rust...**
- **Efficient and Fast**
- **Safe**

WASM format

DEV CAFE
Connect and share

TEXTUAL (.WAT)

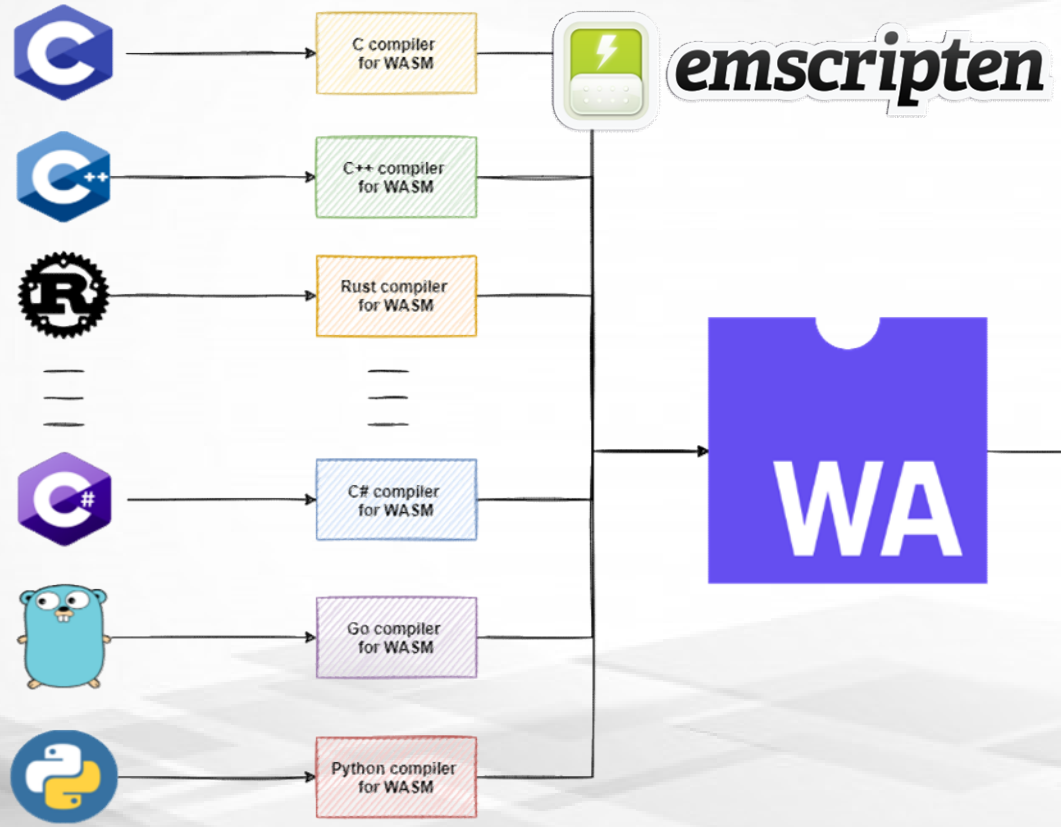
```
(module
 (table 0 anyfunc)
 (memory $0 1)
 (export "memory" (memory $0))
 (export "_Z3Sumii" (func $_Z3Sumii))
 (func $_Z3Sumii (; 0 ;) (param $0 i32) (param $1 i32) (result
   i32)
  (i32.add
   (get_local $1)
   (get_local $0)
  )
 )
)
```

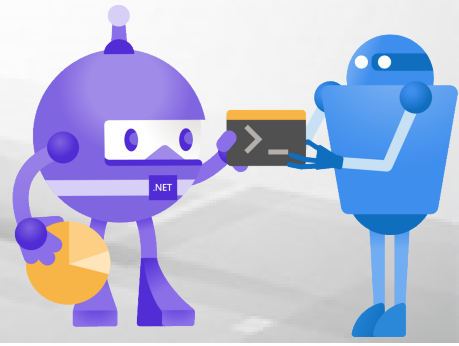
BINARY (.WASM)

```
wasm-function[0]:
  sub rsp, 8                    ; 0x000000 48 83 ec 08
  mov ecx, esi                  ; 0x000004 8b ce
  mov eax, ecx                  ; 0x000006 8b c1
  add eax, edi                  ; 0x000008 03 c7
  nop                           ; 0x00000a 66 90
  add rsp, 8                    ; 0x00000c 48 83 c4 08
  ret                           ; 0x000010 c3
```

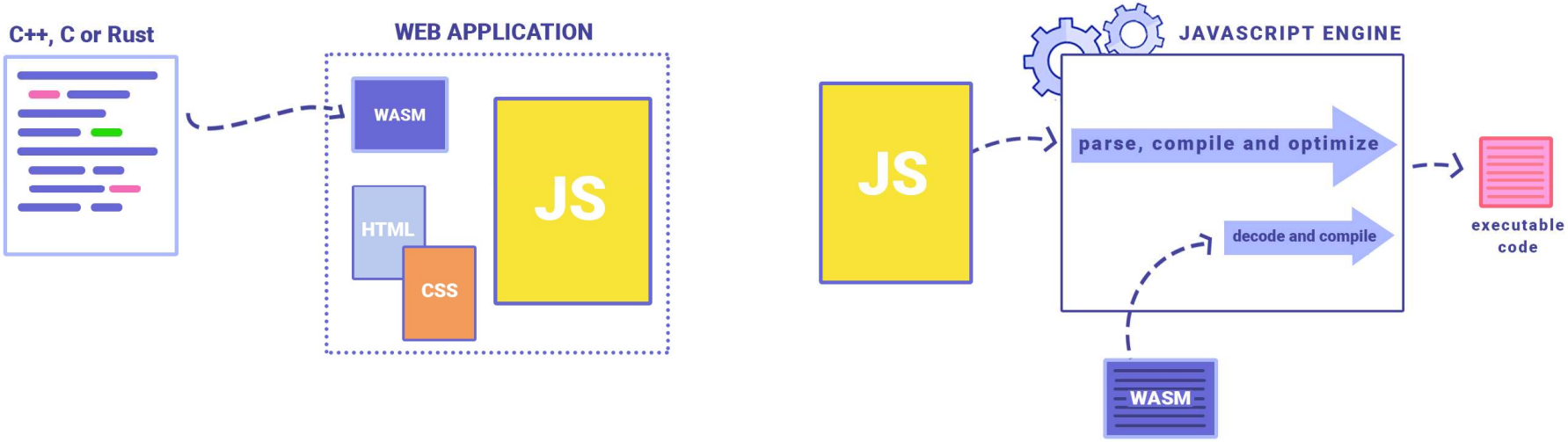
How it works?

DEV CAFE
Connect and share





Why WebAssembly?



- It's not a replacement for JavaScript, it works alongside JavaScript.
- Better performance than JavaScript

DEV CAFE
Connect and share

DEV CAFE
Connect and share

Blazor Overview

Blazor

Blazor is a client-side library that uses .NET on Web Assembly to support Single Page Applications written in C# using Razor templates.

- Combination of .NET Code and Razor syntax
- Free and Open Source
- Not like Silverlight
- Can run without plugins
- Share class libraries between client and server

DEV CAFE
Connect and share

Web Development

DEV CAFE
Connect and share

in the Browser

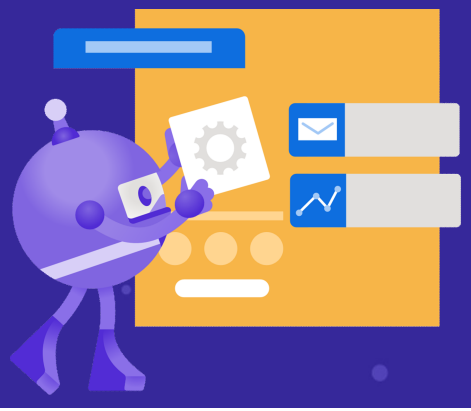
History

DEV CAFE
Connect and share

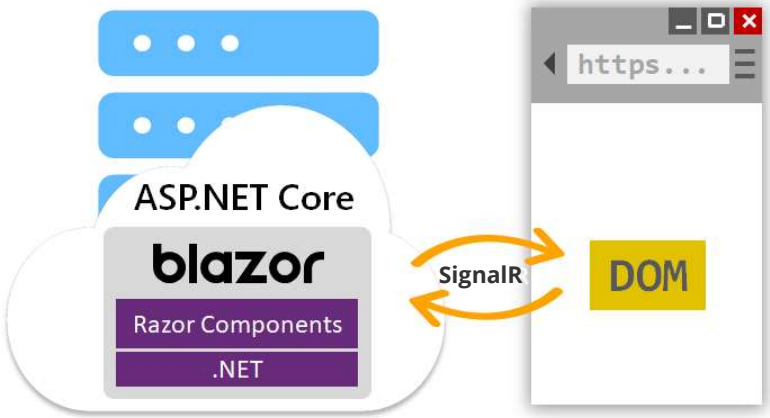
- 2006 – Silverlight
- 2011 – Silverlight 5 (RIP)
- 2017 – Web Assembly
- 2017 - Blazor was announced (Steve Sanderson)
- 2019 - Blazor Preview (Daniel Roth)
- 2019 – Blazor Server - .NET Core 3.0 (Sept)
- 2019 – Blazor WebAssembly - .NET Core 3.1 (Dec)
- 2021 – Blazor LTS - .NET 6 (Nov)

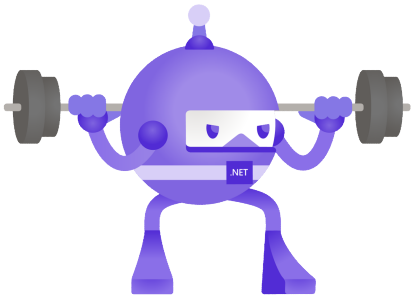
DEV CAFE
Connect and share

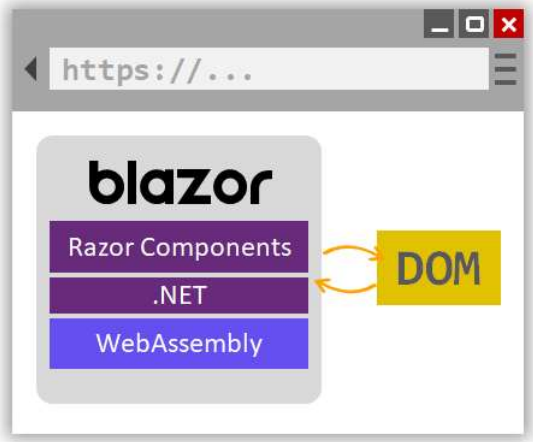
Blazor Server & Blazor WebAssembly



Blazor Hosting Models







Blazor Server

- Server-side logic and rendering
- DOM updates via SignalR

Blazor WebAssembly

- Client-side logic
- Incremental DOM

DEV CAFE
Connect and share

Blazor Server

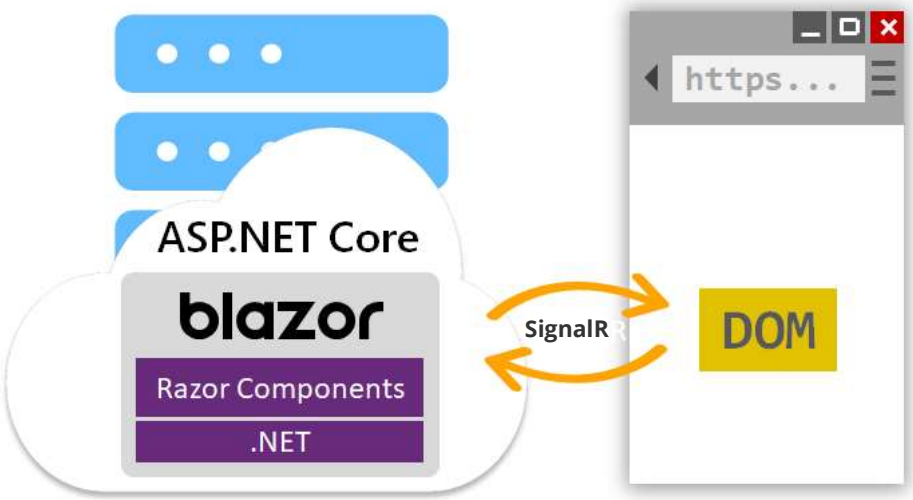
DEV CAFE
Connect and share

❖ PROS

- ✓ No initial download size
- ✓ Server-side rendering
- ✓ Faster in the first load time
- ✓ Work on older browsers
- ✓ API/Server code is private

❑ CONS

- ➢ Latency – SignalR on each event
- ➢ Maintain client connections
- ➢ No offline support
- ➢ Maintain client state
- ➢ High memory usage



Blazor WebAssembly

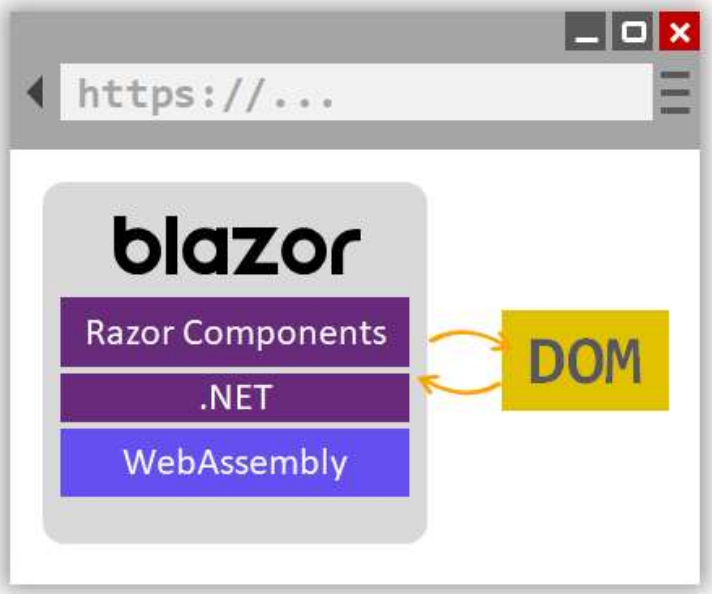
DEV CAFE
Connect and share

❖ **PROS**

- ✓ .NET Code in the browser
- ✓ “Near” native speed
- ✓ Shared code
- ✓ No server-side dependency
- ✓ Progressive Web App (PWA)

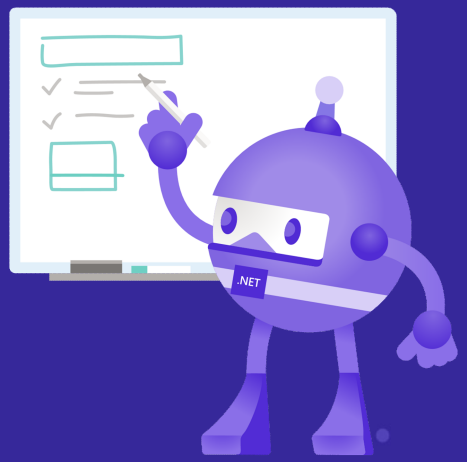
❑ **CONS**

- ➢ Initial download size (~700kb)
- ➢ Need a modern browser
- ➢ Tooling still need to mature



DEV CAFE
Connect and share

Demo



Component Anatomy

DEV CAFE
Connect and share

Directives

```
@page "/"
@using BlazorSignalRApp.Shared;
@inject HttpClient Http
```

Markup & Razor

```
<h1>@PageTitle</h1>

@if (forecasts == null)
{
    <p><em>Loading...</em></p>
} ...
```

Code/Logic

```
@code {
    private string PageTitle = "Blazor Weather App";
    WeatherForecast[] forecasts;

    protected override async Task OnInitializedAsync()
    {
        forecasts = await Http.GetFromJsonAsync<WeatherForecast[]>("WeatherForecast");
    }
}
```

References

- Awesome WebAssembly List: https://github.com/mbasso/awesome-wasm
- Awesome Blazor List: https://github.com/AdrienTorris/awesome-blazor

Thanks for joining!

letienthanh0212@gmail.com