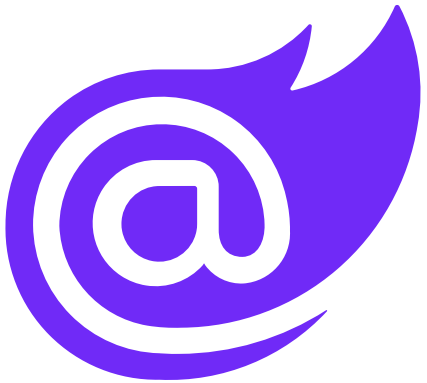
Blazor

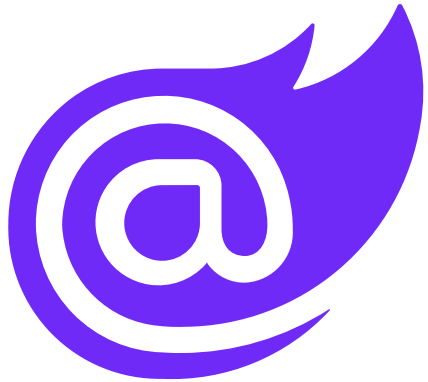
new way to build client web apps



Thien Nguyen

Agenda

1. What is WebAssembly
2. What is Blazor
3. Blazor hosting models
4. PROS and CONS of each Blazor hosting models
5. Demo

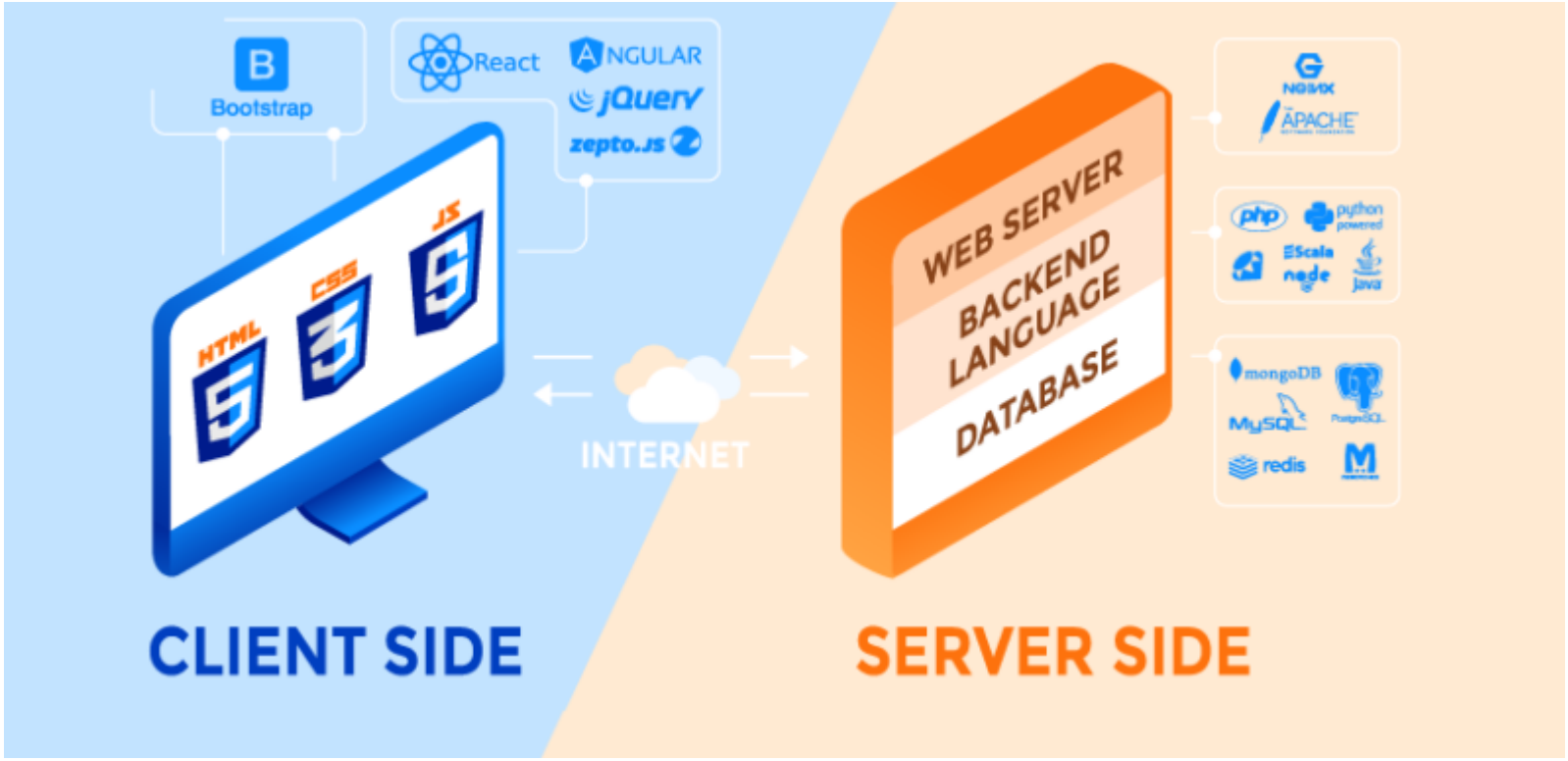


What is WebAssembly

1. What is WebAssembly

1. What is WebAssembly

Review about Server-Side and Client-Side Programming Languages



1. What is WebAssembly

- WebAssembly (abbreviated *Wasm*) is a binary instruction format for a stack-based virtual machine

- Wasm is designed as a portable compilation target for programming languages, enabling deployment on the web for client and server applications.

- Wasm became the offical ***"fourth language for web"*** on 5th December 2019, i.e. after HTML, CSS & JavaScript

- Can written by C/C++, C#, Python,…

https://webassembly.org/

1. What is WebAssembly

WASM format

textual format
.wat

binary format
.wasm

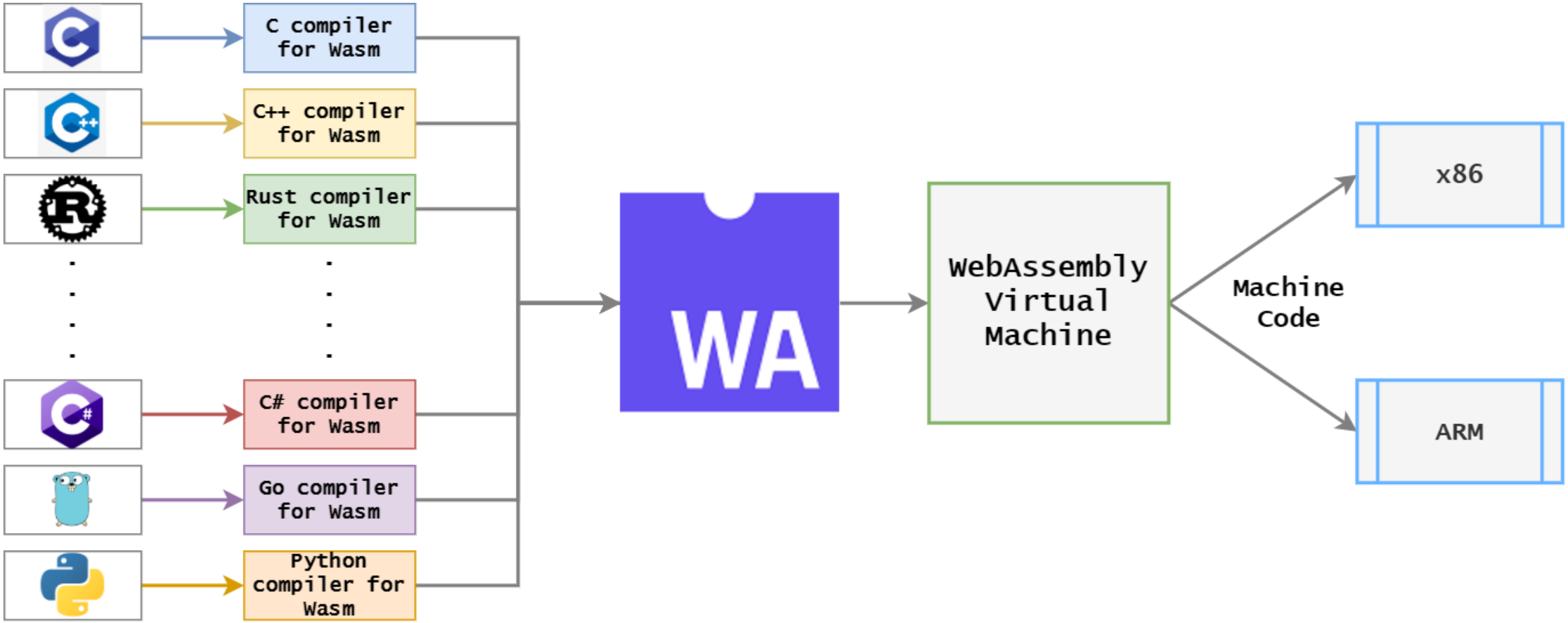
1. What is WebAssembly

WASM format

me* after reading the WASM formats the first time

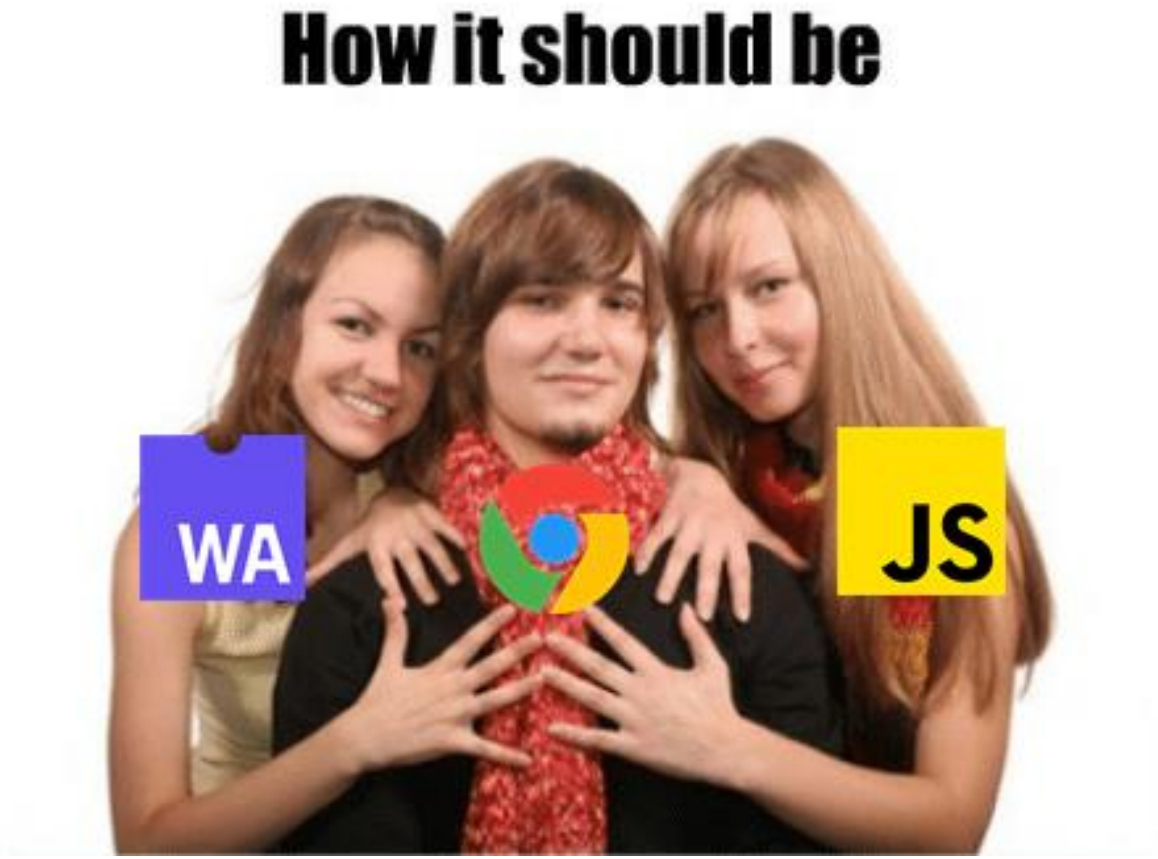
1. What is WebAssembly

WebAssembly Work



1. What is WebAssembly

Why would I use WebAssembly?



1. What is WebAssembly

Why would I use WebAssembly?

- ❑ Efficient and fast
- ❑ Safe
- ❑ Compile a WebAssembly module from many languages: C/C++, Rust, C#, F#, Go, Kotlin, Swift,…
- ❑ Can be executed on web, stand-alone

1. What is WebAssembly

Besides, WebAssembly not-so-great

- ❑ Many features under development or not ready yet
- ❑ IE and older browsers don’t support wasm
- ❑ By design, code needed to be compiled before distribution. Therefore, “***fix JS directly on CDN***” kind of approaching doesn’t supported

What is Blazor

2. What is Blazor

- Release in 2018
- A client-side library that use .NET on WebAssembly
- Support SPAs written in C# with Razor template

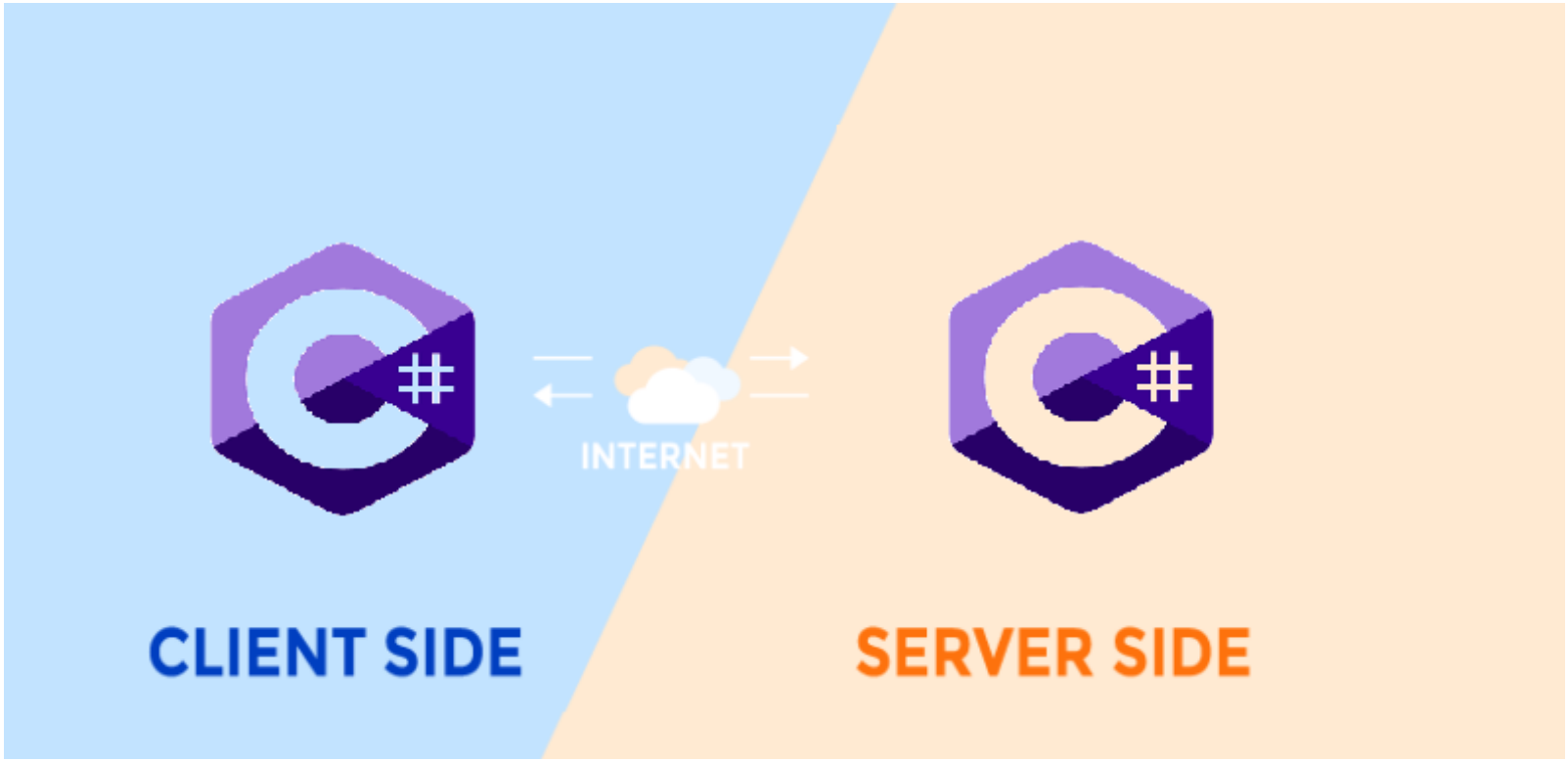
https://docs.microsoft.com/en-us/aspnet/core/blazor/?view=aspnetcore-6.0

2. What is Blazor

- ❑ Not like Silverlight
- ❑ Can run without plugin
- ❑ Share libraries between client and server

2. What is Blazor

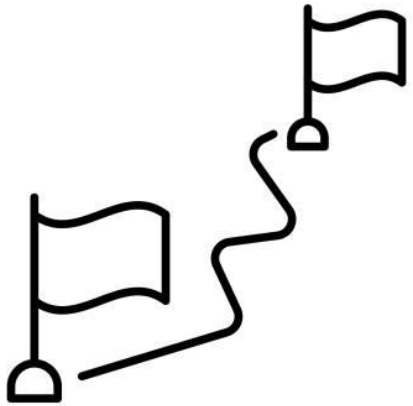
Server-Side and Client-Side Programming Languages



2. What is Blazor

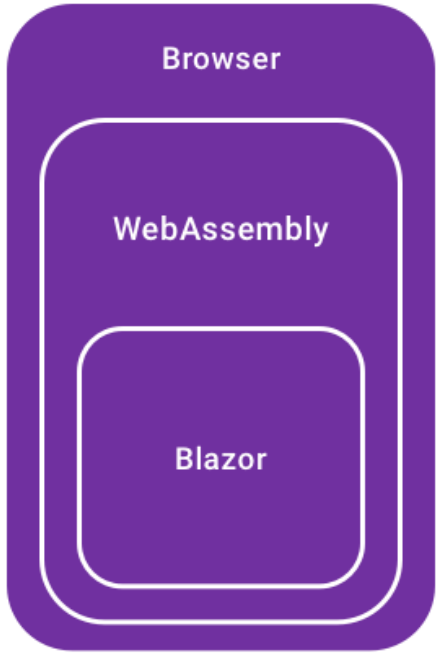
Milestones

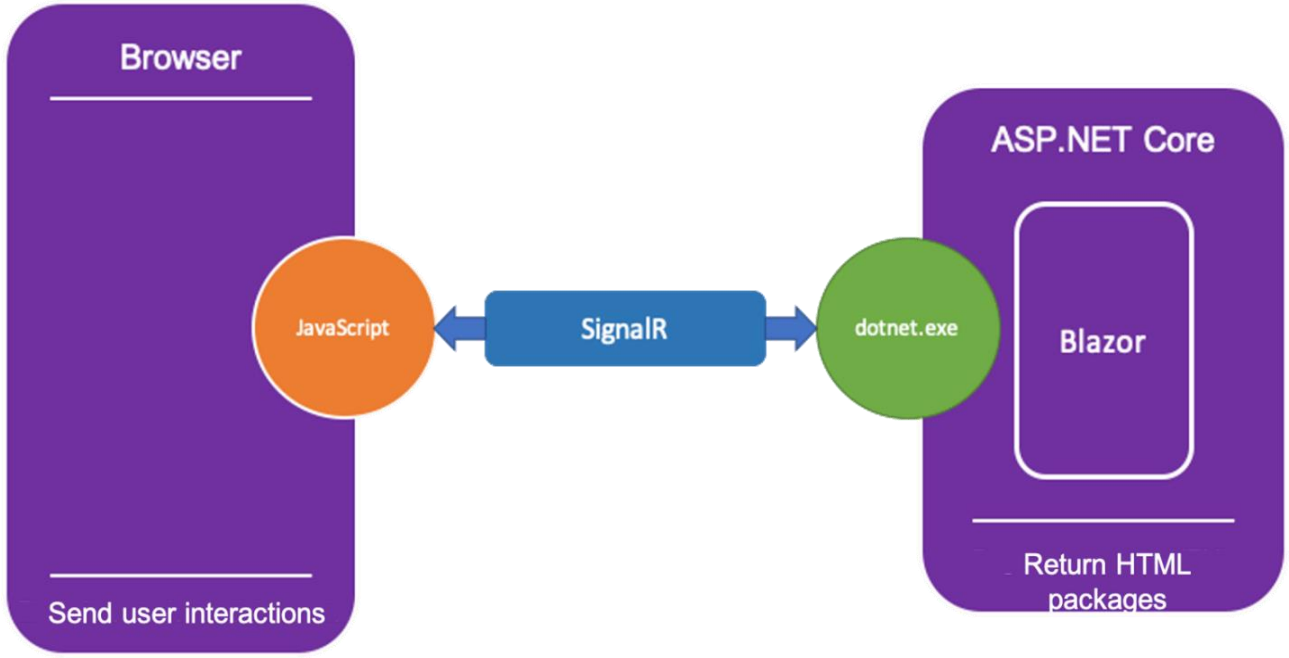
- 2017 : Web Assembly
- 2017 : Blazor 1st announced
- 2019 : Blazor 1st preview
- 2019 : Blazor Server
- 2019 : Blazor Web Assembly
- 2021 : LTS for Blazor
- 2022 : .NET 7



Blazor hosting models

3. Blazor hosting models





Blazor WebAssembly

- ❑ Client-side rendering
- ❑ Incremental DOM

Blazor Server

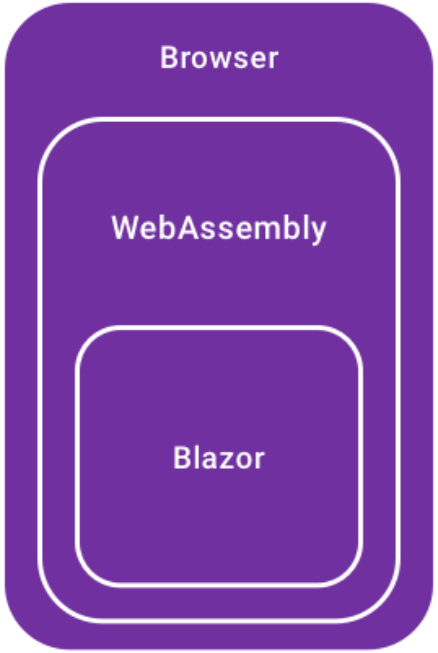
- ❑ Server-side logic
- ❑ DOM updated by SignalR

PROS and CONS

Blazor WebAssembly and Blazor Server

4. PROS and CONS Blazor WebAssembly and Blazor Server

Blazor WebAssembly



PROS

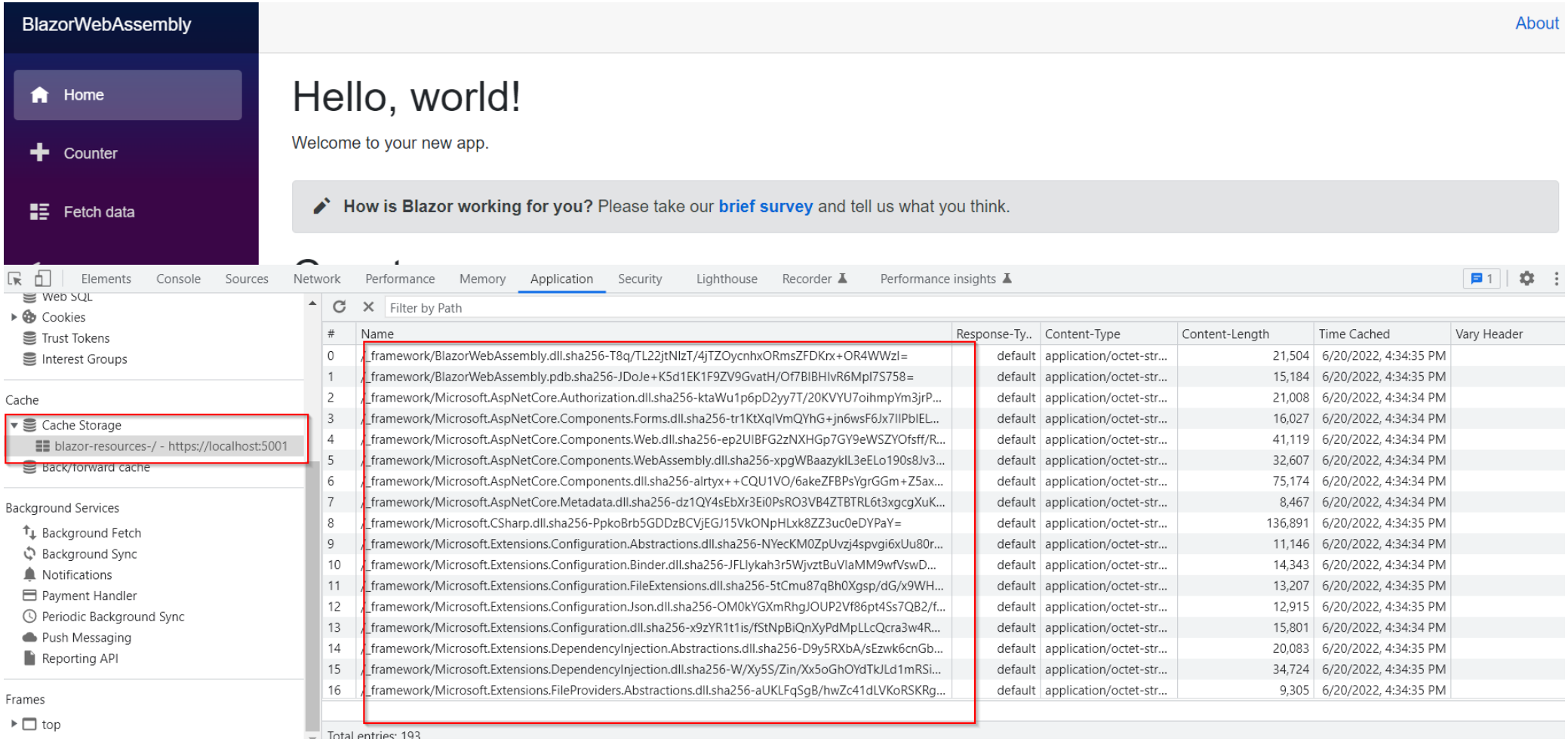
- ❑ .NET code for browser
- ❑ Efficient and fast
- ❑ NO server-side dependencies
- ❑ Progressive Web App (PWA)

CONS

- ❑ Need download resources before 1st running (~700 KB)
- ❑ Old browser cannot be used
- ❑ Need times to developing tools for developers

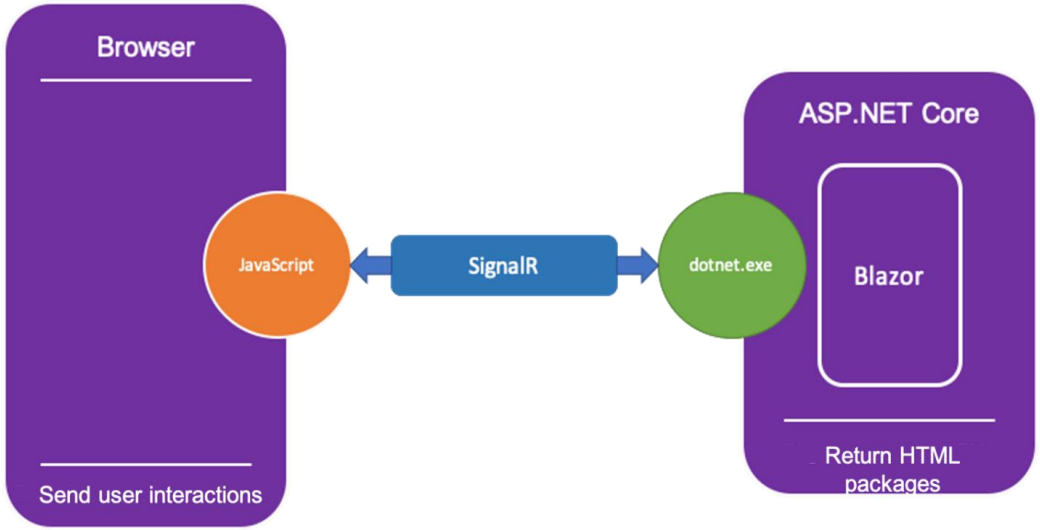
4. PROS and CONS Blazor WebAssembly and Blazor Server

Blazor WebAssembly



4. PROS and CONS Blazor WebAssembly and Blazor Server

Blazor Server



PROS

- ❑ NO need to download before running
- ❑ Server-side rendering
- ❑ Load fast in 1st time
- ❑ Can be used in older browser
- ❑ API private

CONS

- ❑ SignalR -> latency on each event
- ❑ Always keeps connection
- ❑ NO offline mode
- ❑ High memory usage

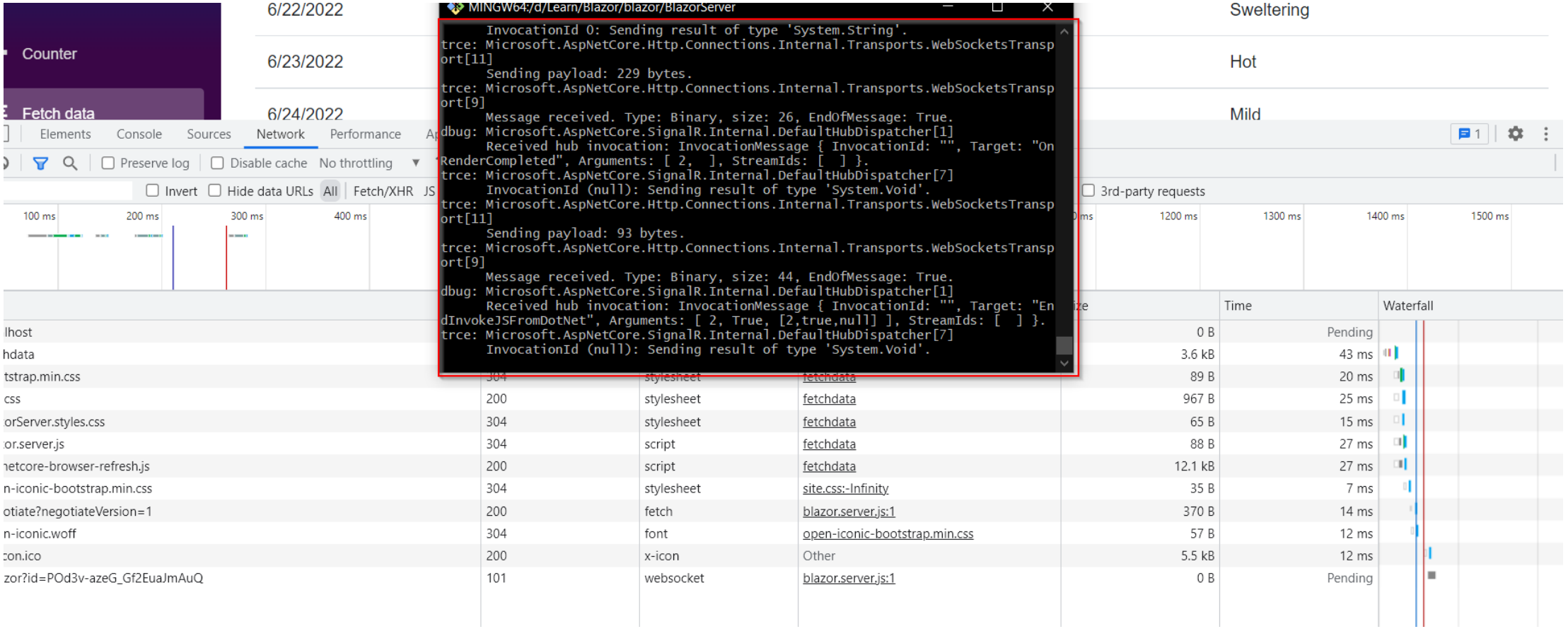
4. PROS and CONS Blazor WebAssembly and Blazor Server

Blazor Server

```
19      public static IHostBuilder CreateHostBuilder(string[] args) =>
20          Host.CreateDefaultBuilder(args)
21          .ConfigureLogging(logging =>
22          {
23              logging.ClearProviders();
24              logging.AddConsole();
25              logging.AddFilter(
26                  "Microsoft.AspNetCore.SignalR", LogLevel.Trace);
27              logging.AddFilter(
28                  "Microsoft.AspNetCore.Http.Connections",
29                  LogLevel.Trace);
30          })
31          .ConfigureWebHostDefaults(webBuilder =>
32          {
33              webBuilder.UseStartup<Startup>();
34          });
35      }
36  }
```

4. PROS and CONS Blazor WebAssembly and Blazor Server

Blazor Server



Demo

References

- Server-Side vs Client-Side Programming Languages
https://www.techwebspace.com/server-side-vs-client-side-programming-languages/

- WebAssembly
https://arghya.xyz/articles/webassembly-wasm-wasi/
https://developer.mozilla.org/en-US/docs/WebAssembly/Understanding_the_text_format

- Blazor
https://docs.microsoft.com/en-us/aspnet/core/blazor/hosting-models?view=aspnetcore-5.0

Some relevant questions

- ASP.NET Core hosted vs Blazor Serverside
https://stackoverflow.com/questions/58093386/whats-the-difference-between-asp-net-core-hosted-and-server-side-blazor-really

- JS vs WASM
https://medium0.com/@OPTASY.com/webassembly-vs-javascript-is-wasm-faster-than-js-when-does-javascript-perform-better-db86d2ecf2cc

Thank you for watching