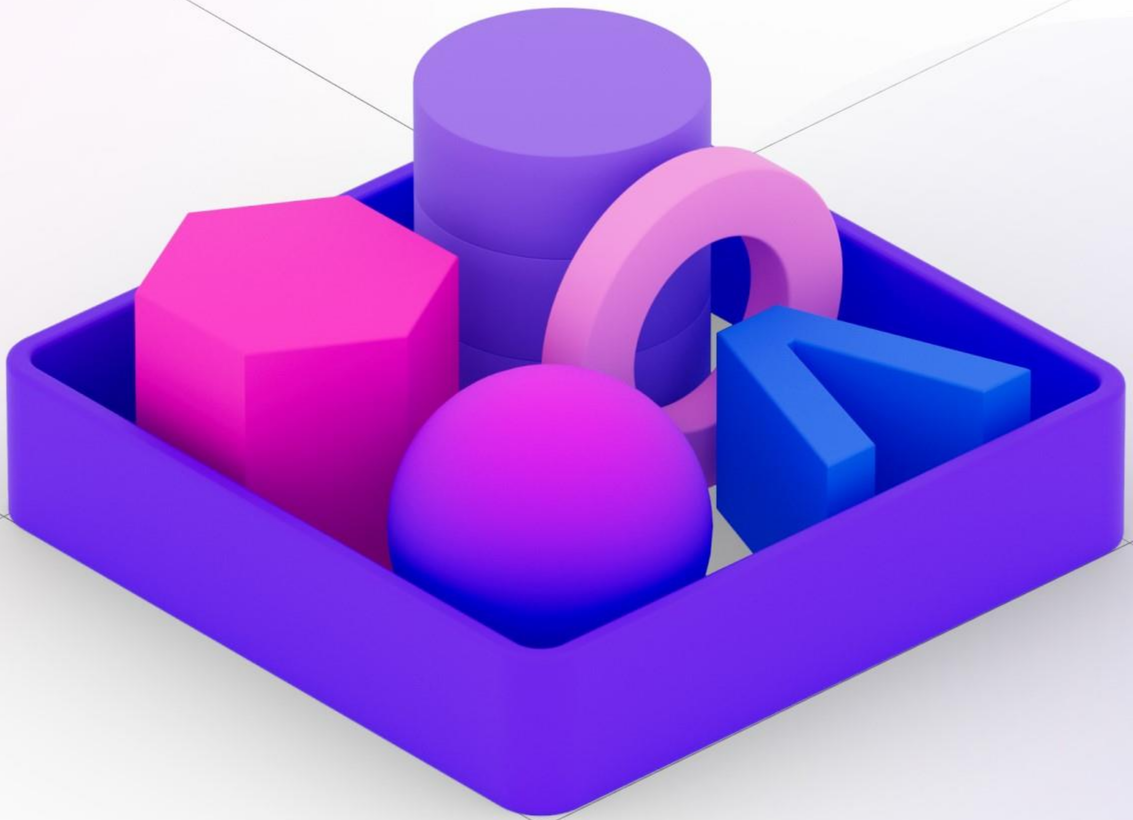
.NET Conf 2023 @ Blazor Korea

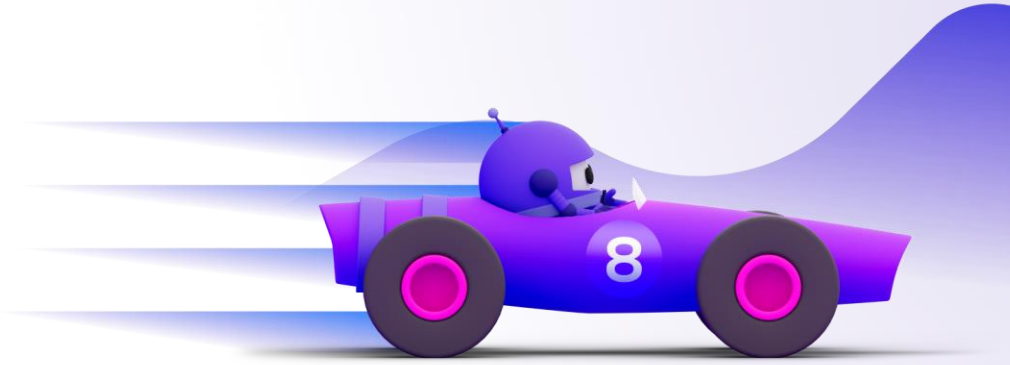


새로워진 Blazor

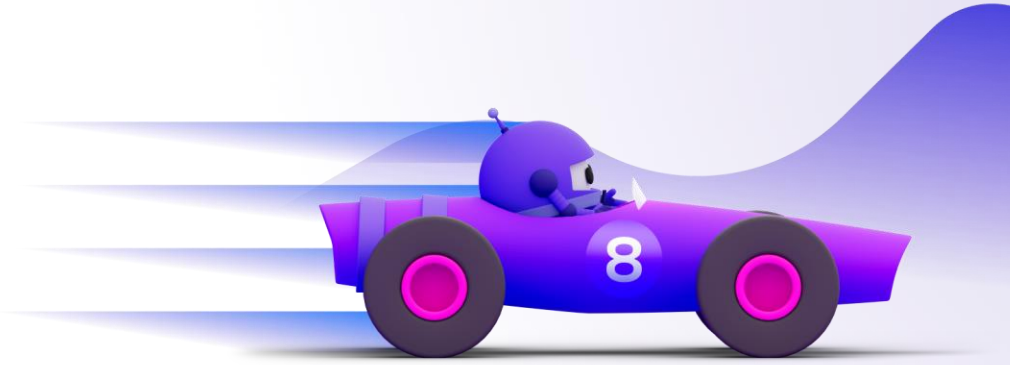
Speaker

- 김 진석
- **(주)케이에이치시스템즈** 대표이사
- 한국블레이저개발자모임 / Blazor Korea
- facebook.com/iamjinseok.kr

Blazor ?



블레이저!



Blazor?

- .NET 및 C#의 기능을 사용하여
- JavaScript를 작성하지 않고
- 전체 스택 웹 앱(Full Stack Web App)을 빌드

https://dotnet.microsoft.com/ko-kr/apps/aspnet/web-apps/blazor

Blazor?

Blazor 기술 배경

- C# and .NET
- Single Page Application
- Minimal API
- Signal R
- WebAssembly
- BootStrap

https://dotnet.microsoft.com/ko-kr/apps/aspnet/web-apps/blazor

Blazor?

C# and .NET

Microsoft
.NET

C#

- 객체지향 프로그래밍 언어
- 문법적인 특성이 JAVA와 유사
- C# 12

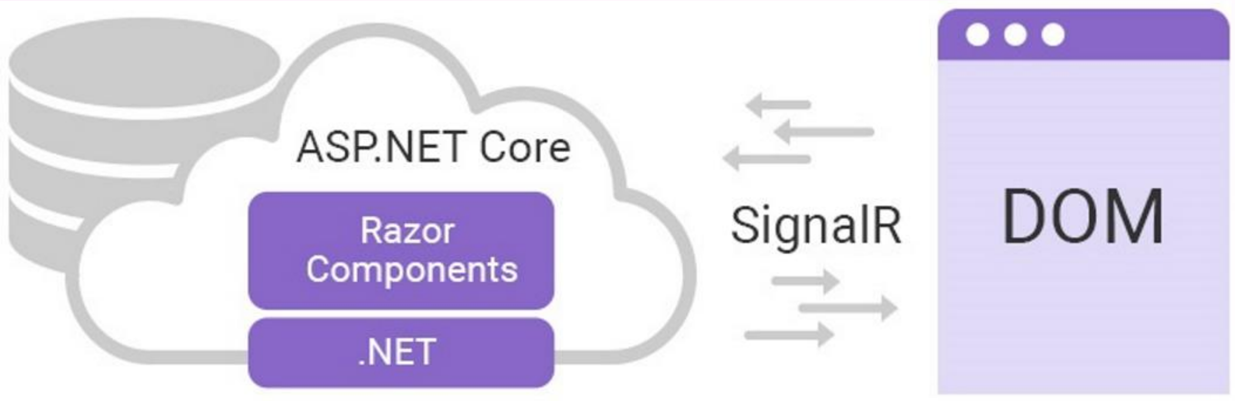
.NET

- 오픈소스 개발자 플랫폼
- 웹, 모바일, 데스크톱, 게임, IoT 등을 빌드
- 크로스플랫폼(일부)

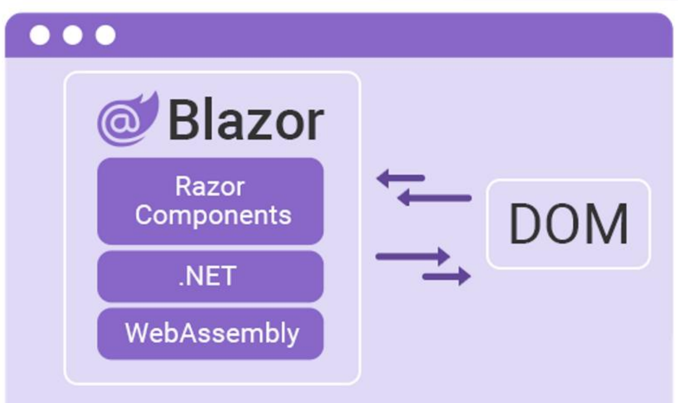
https://dotnet.microsoft.com/ko-kr/apps/aspnet/web-apps/blazor

Blazor?

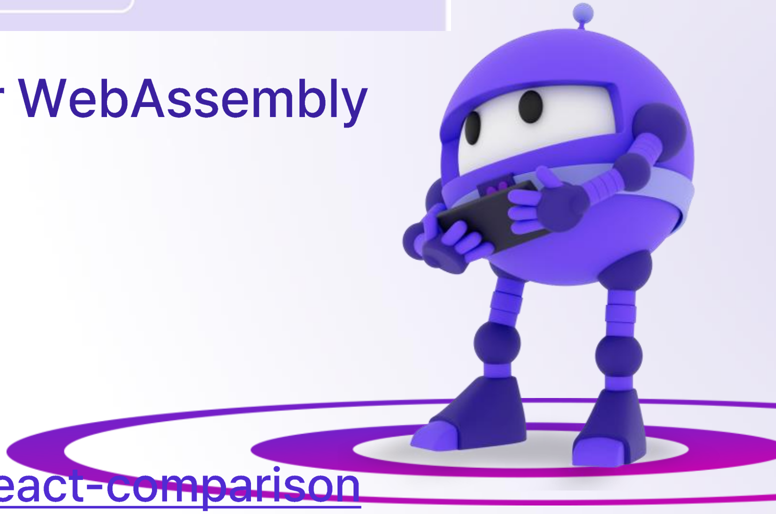
Single Page Application



Blazor Server



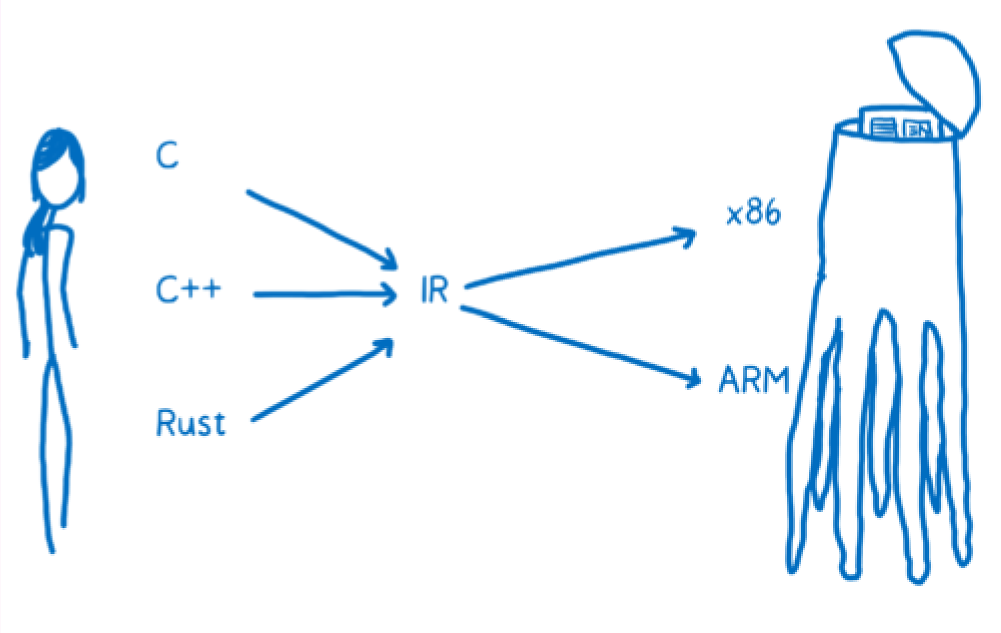
Blazor WebAssembly



https://www.infragistics.com/community/blogs/b/jason_beres/posts/blazor-vs-react-comparison

Blazor?

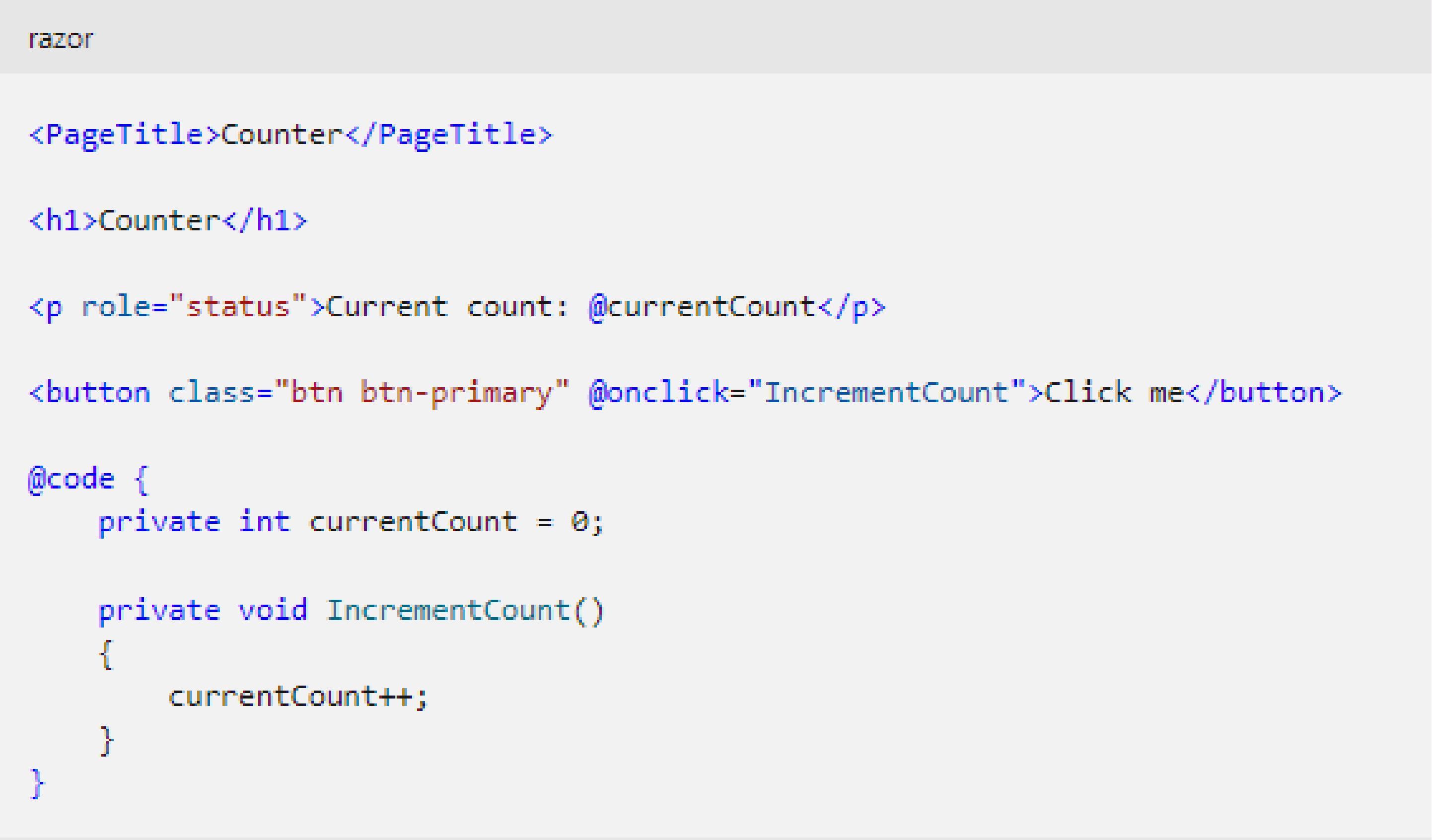
WebAssembly



- 2017년 처음 발표
- 브라우저에서 실행되는 바이트 코드
- 최신 브라우저 지원

Blazor?

code example



razor

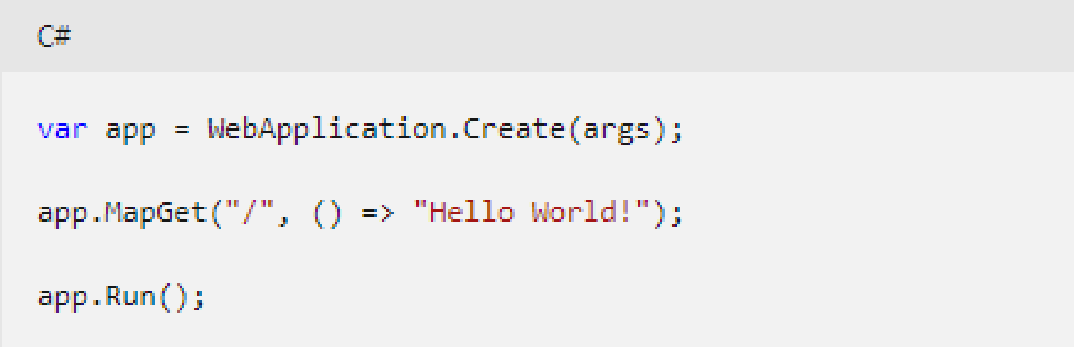
```
<PageTitle>Counter</PageTitle>

<h1>Counter</h1>

<p role="status">Current count: @currentCount</p>

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

@code {
    private int currentCount = 0;

    private void IncrementCount()
    {
        currentCount++;
    }
}
```

- HTML
- @로 시작되는 변수
- C# 코드

https://learn.microsoft.com/en-us/aspnet/core/blazor/

Blazor?

Minimal APIs



C#

```csharp
var app = WebApplication.Create(args);

app.MapGet("/", () => "Hello World!");

app.Run();
```

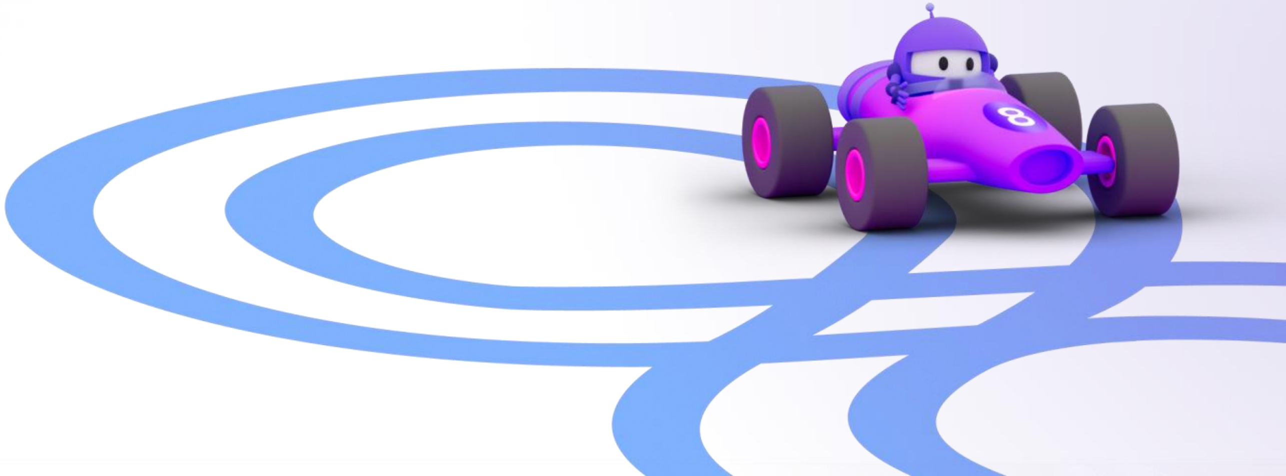
- RESTful API처럼 구현하고자 할 때
- ASP.NET Core의 기능

https://learn.microsoft.com/en-us/aspnet/core/fundamentals/minimal-apis/overview

Blazor?

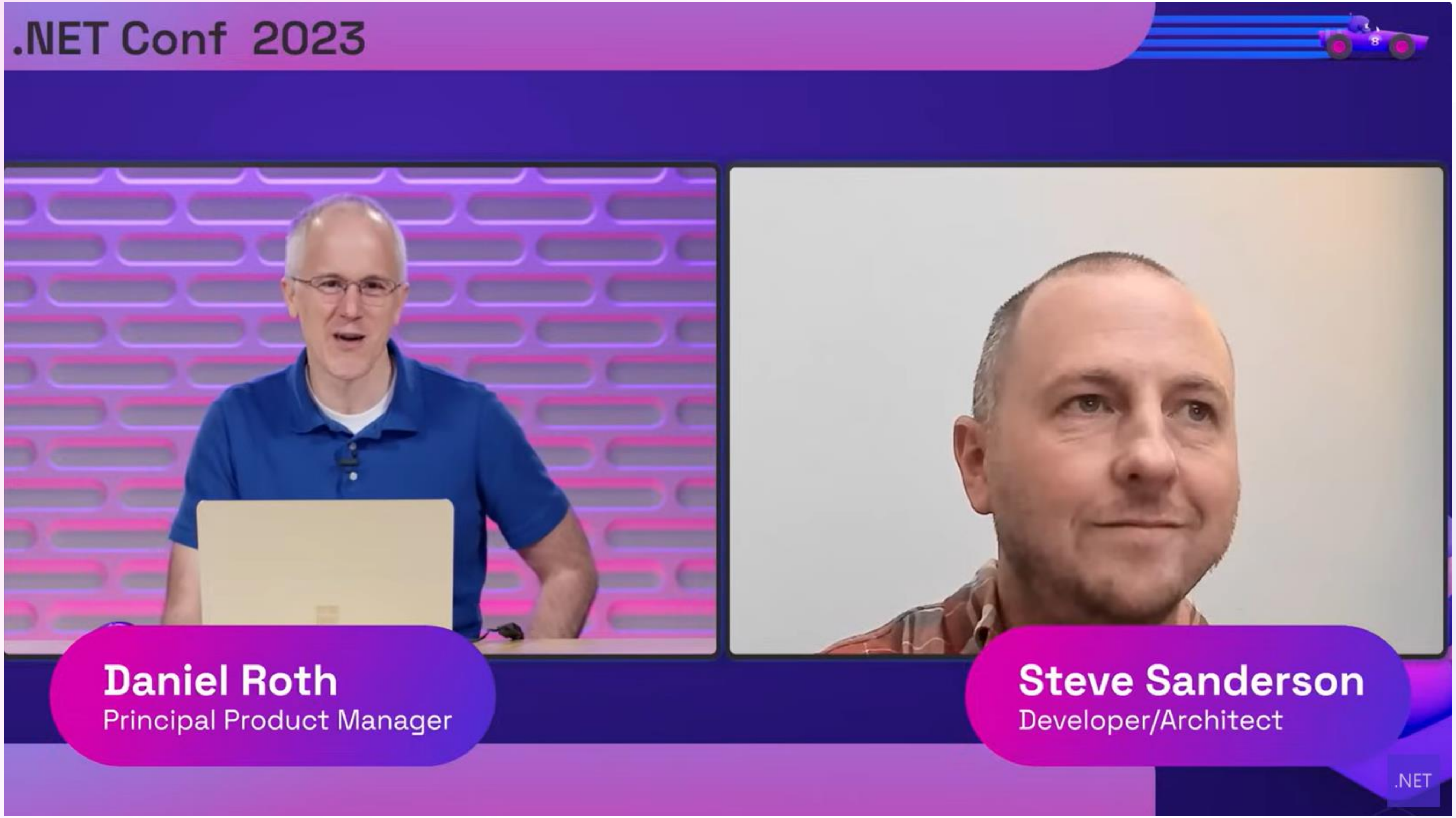
Demo

새로워진 Blazor
Full stack web UI with Blazor in .NET 8

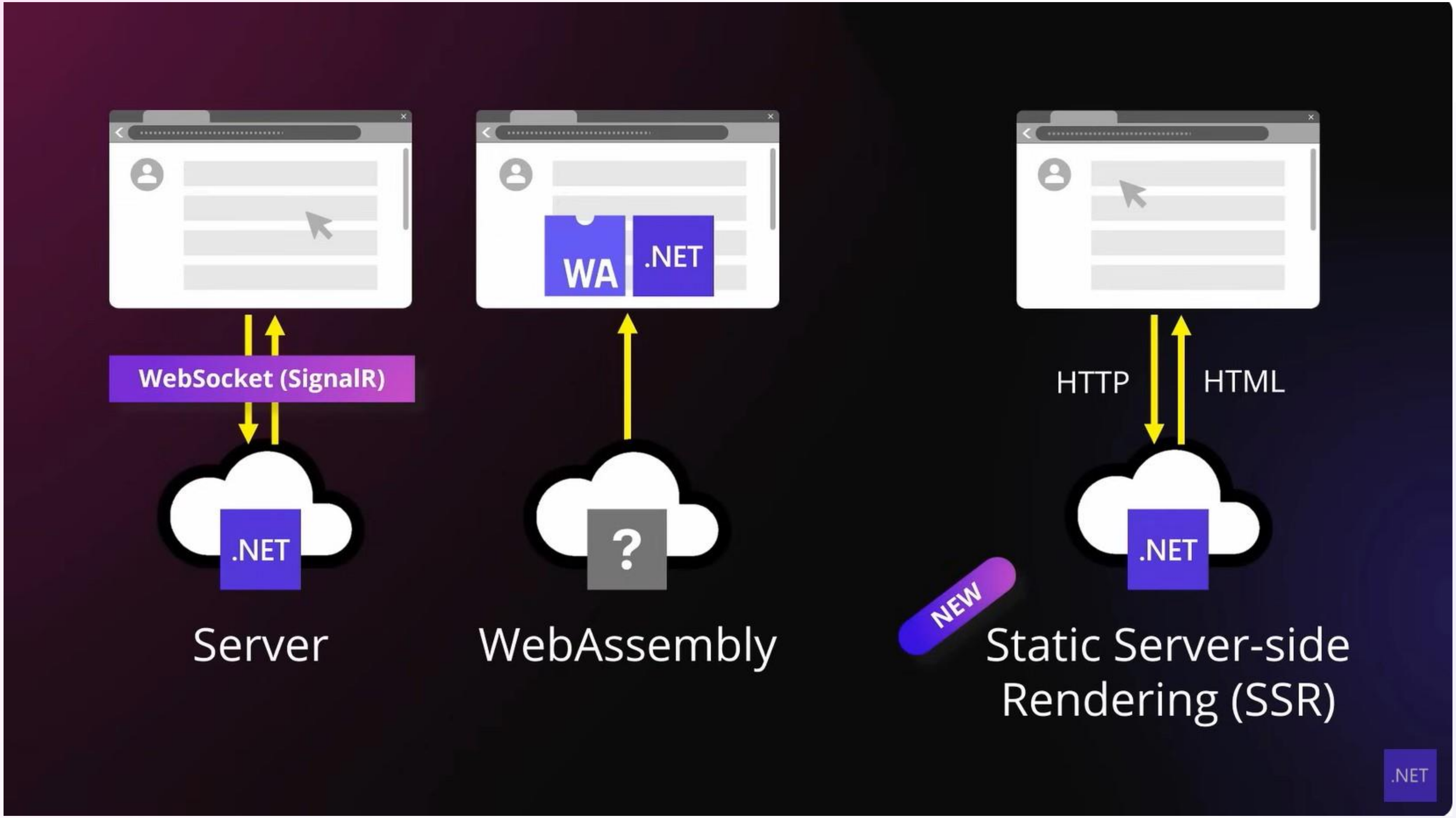


https://www.youtube.com/watch?v=YwZdtLEtROA

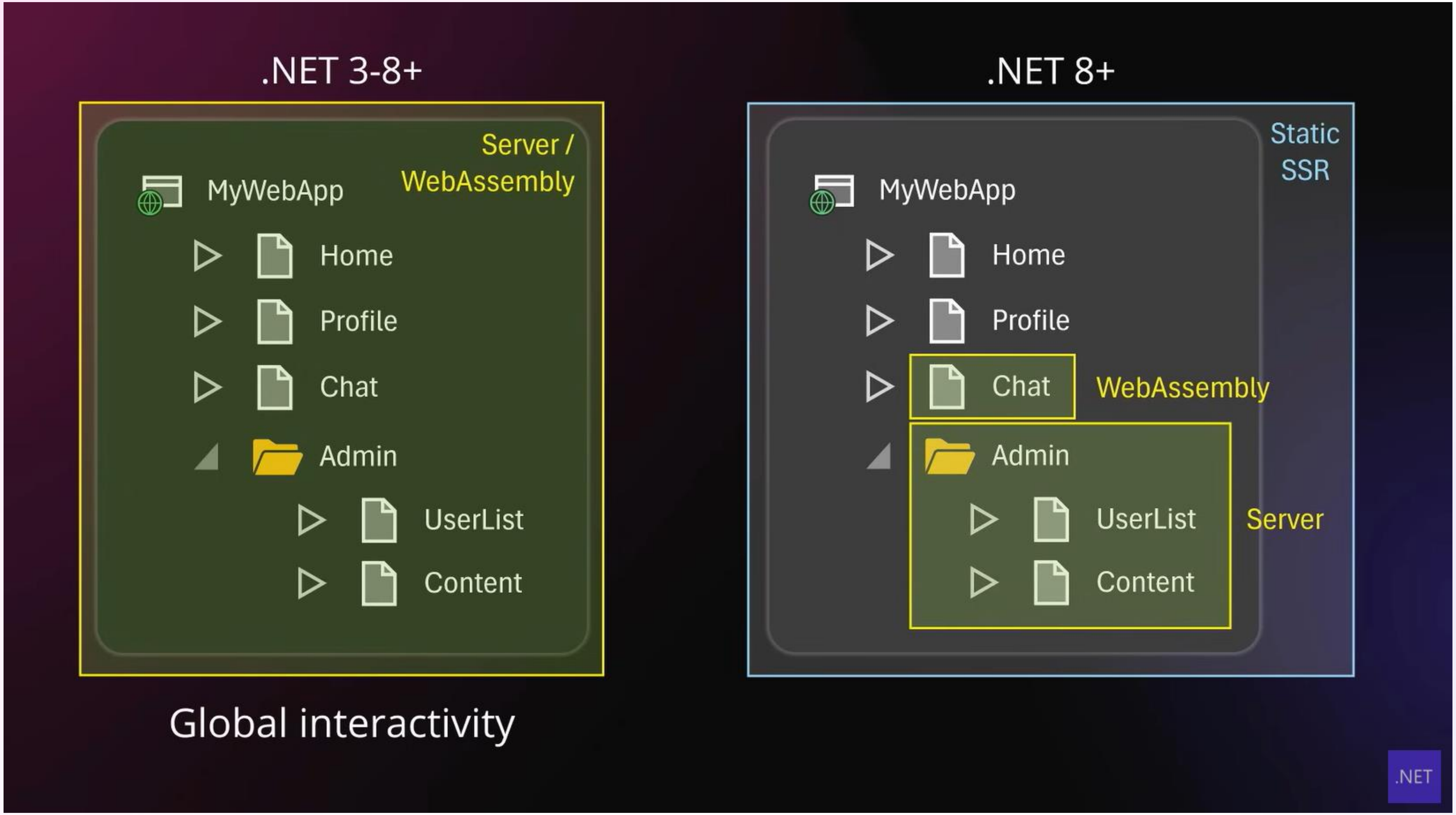
새로워진 Blazor

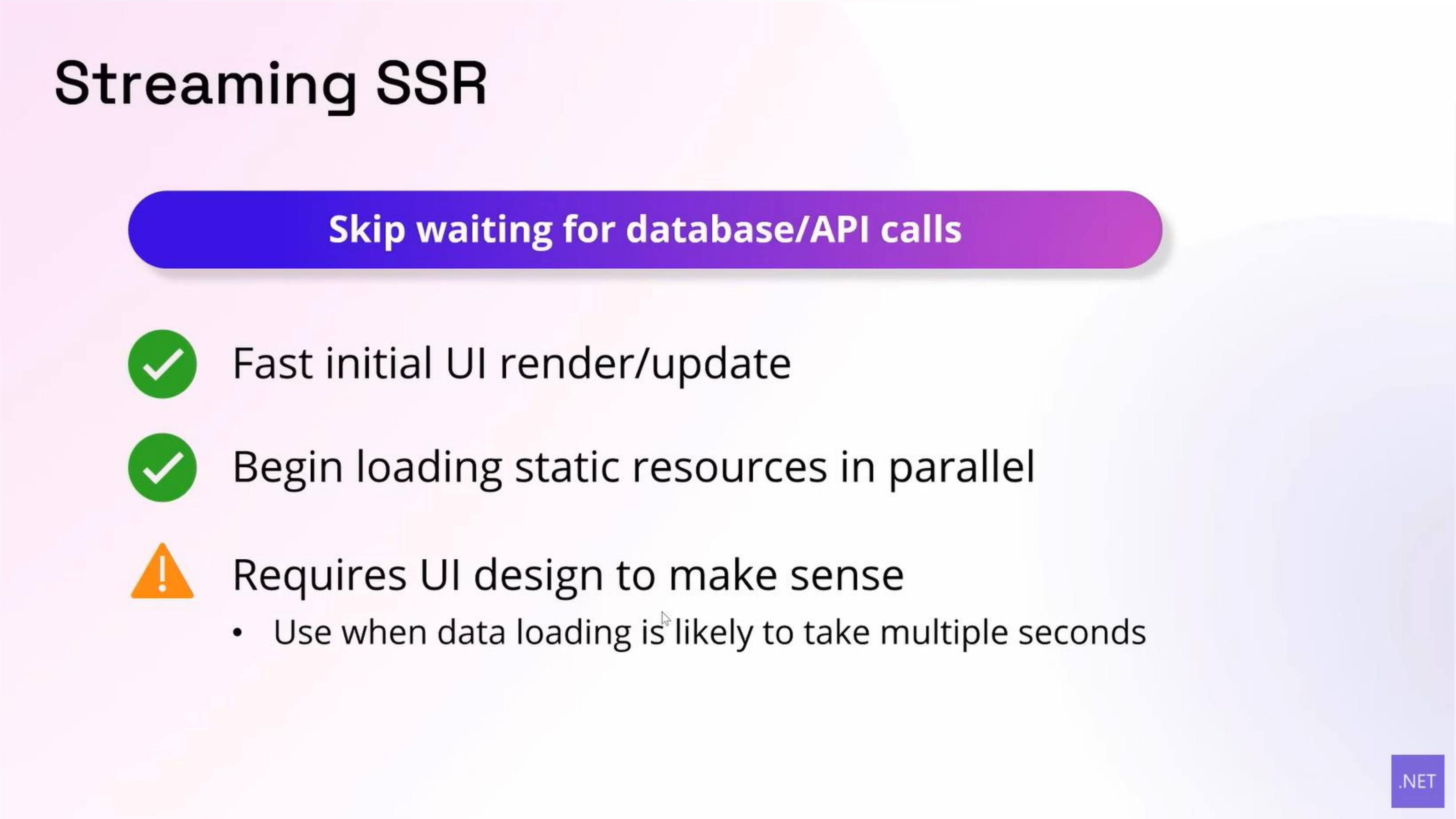

.NET Conf 2023
Daniel Roth
Principal Product Manager
Steve Sanderson
Developer/Architect
.NET

새로워진 Blazor


WA
.NET
WebSocket (SignalR)
.NET
Server
?
WebAssembly
HTTP
HTML
.NET
NEW
Static Server-side Rendering (SSR)
.NET

새로워진 Blazor

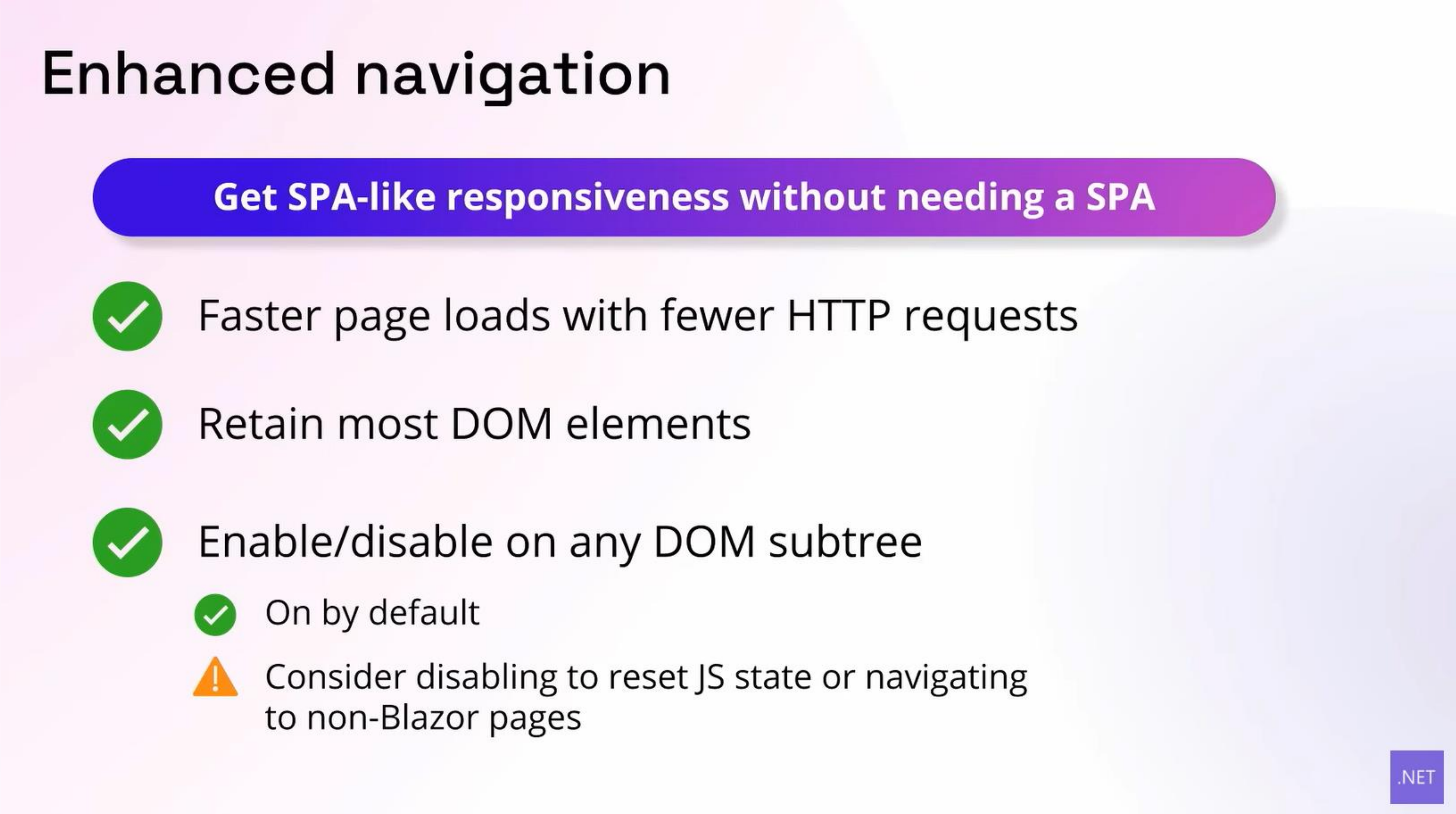

.NET 3-8+
Server / WebAssembly
MyWebApp
Home
Profile
Chat
Admin
UserList
Content
Global interactivity
.NET 8+
Static SSR
MyWebApp
Home
Profile
Chat
WebAssembly
Admin
UserList
Content
Server
.NET



Streaming SSR

Skip waiting for database/API calls

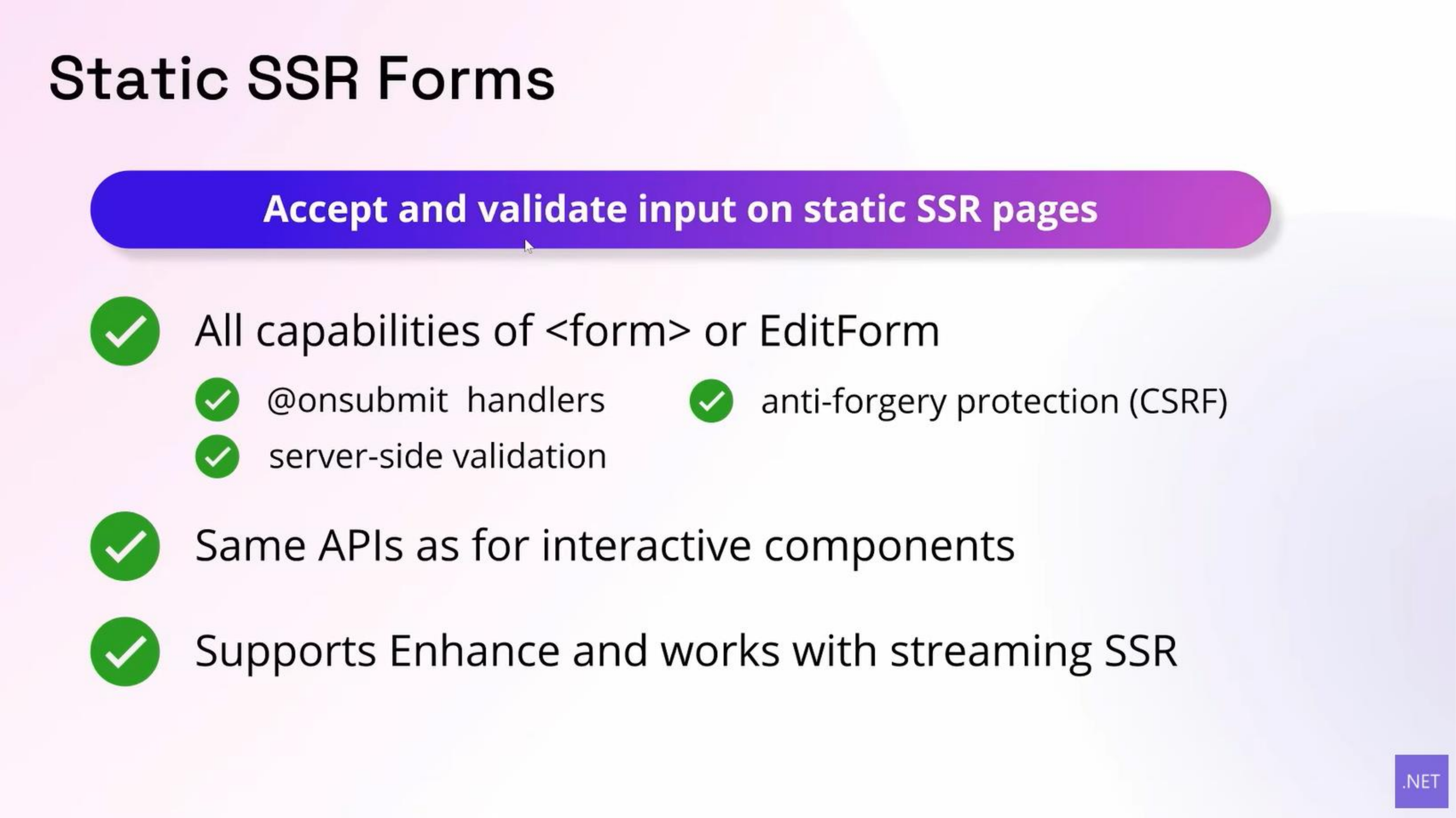
- ✓ Fast initial UI render/update
- ✓ Begin loading static resources in parallel
- ⚠ Requires UI design to make sense
  - Use when data loading is likely to take multiple seconds



Enhanced navigation

Get SPA-like responsiveness without needing a SPA

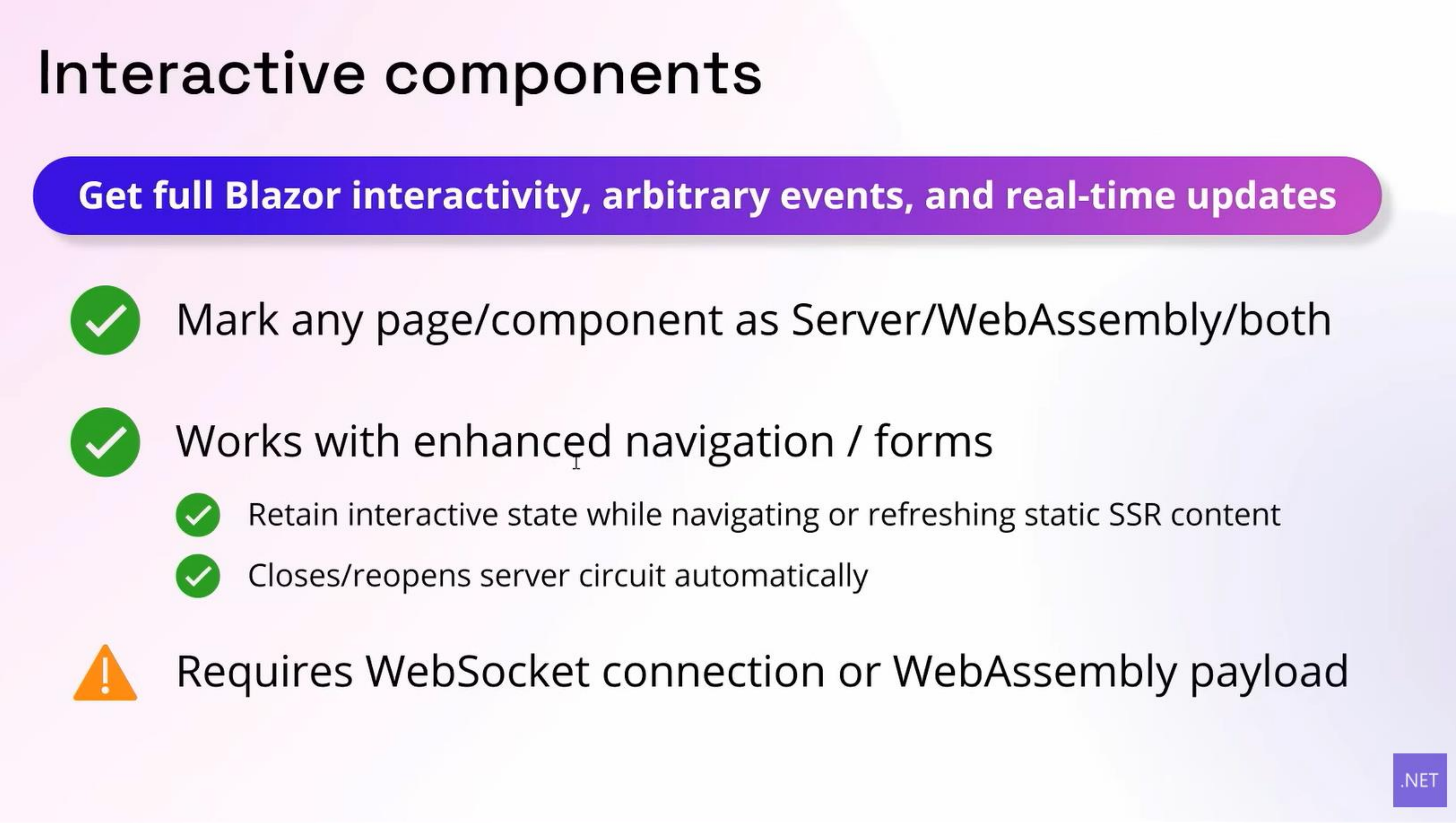
- ✓ Faster page loads with fewer HTTP requests
- ✓ Retain most DOM elements
- ✓ Enable/disable on any DOM subtree
  - ✓ On by default
  - ⚠ Consider disabling to reset JS state or navigating to non-Blazor pages



Static SSR Forms

Accept and validate input on static SSR pages

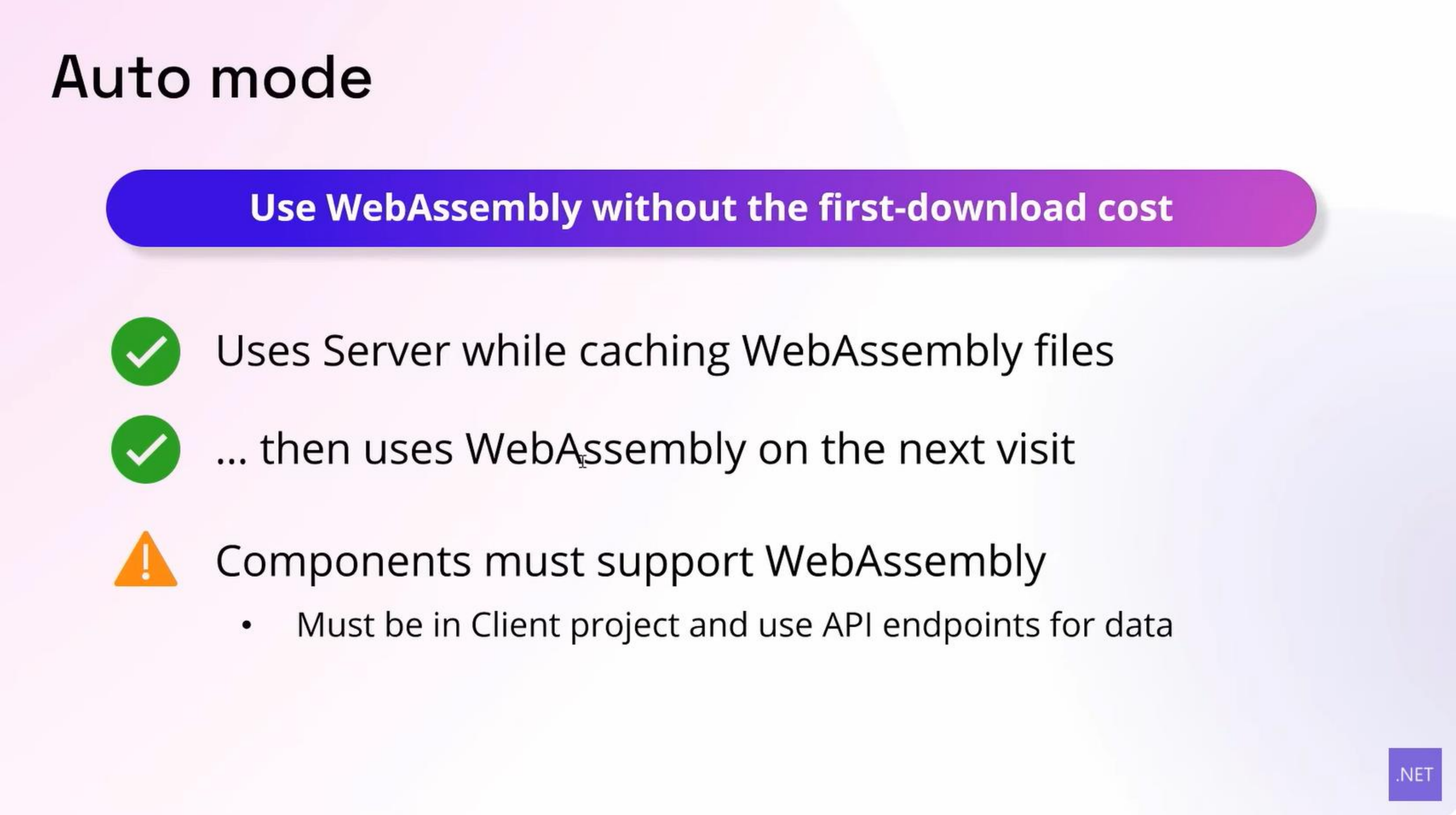
- ✓ All capabilities of <form> or EditForm
  - ✓ @onsubmit handlers
  - ✓ anti-forgery protection (CSRF)
  - ✓ server-side validation
- ✓ Same APIs as for interactive components
- ✓ Supports Enhance and works with streaming SSR



Interactive components

Get full Blazor interactivity, arbitrary events, and real-time updates

- ✓ Mark any page/component as Server/WebAssembly/both
- ✓ Works with enhanced navigation / forms
  - ✓ Retain interactive state while navigating or refreshing static SSR content
  - ✓ Closes/reopens server circuit automatically
- ⚠ Requires WebSocket connection or WebAssembly payload



Auto mode

Use WebAssembly without the first-download cost

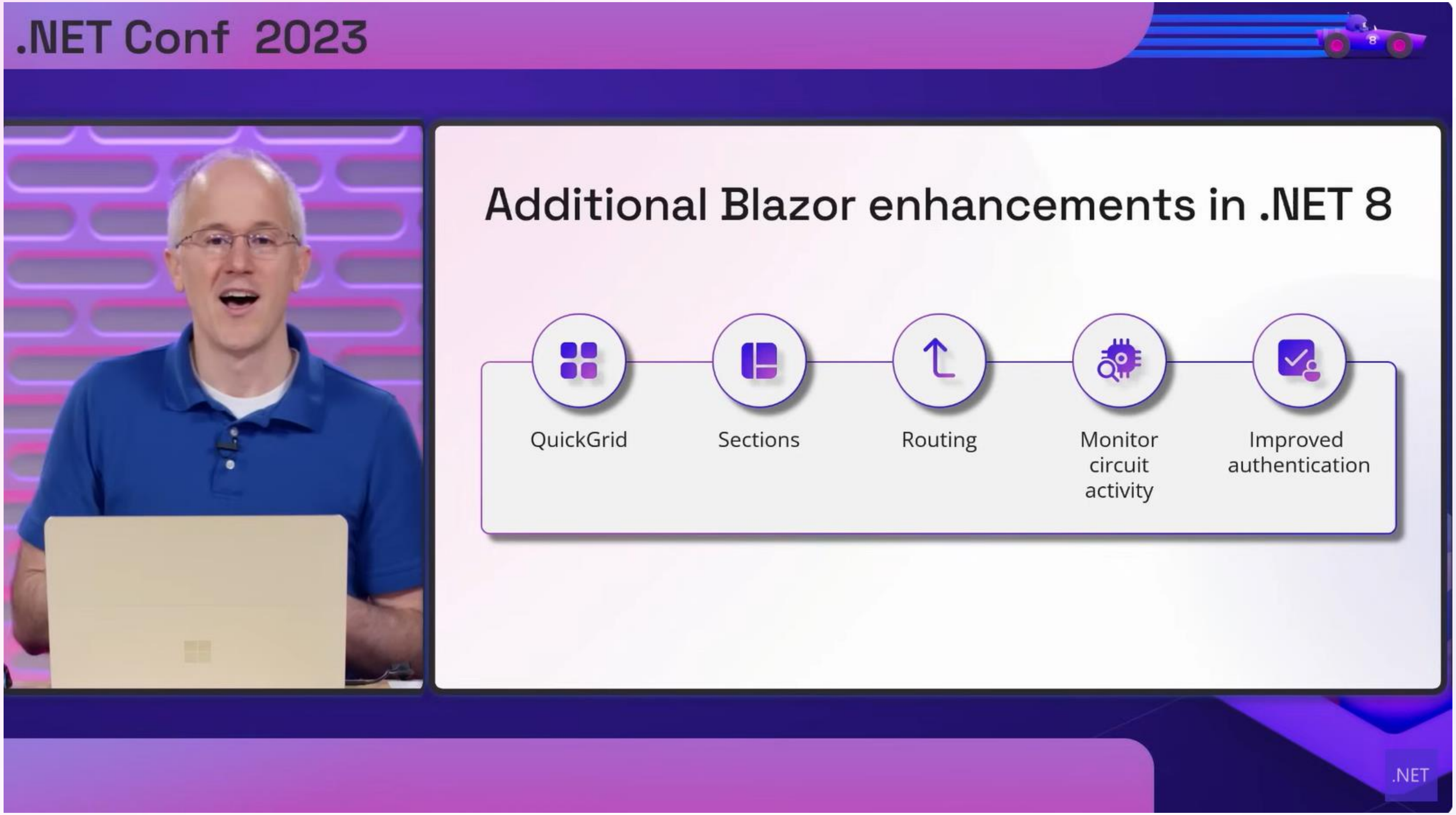
✓ Uses Server while caching WebAssembly files

✓ ... then uses WebAssembly on the next visit

⚠ Components must support WebAssembly

- Must be in Client project and use API endpoints for data

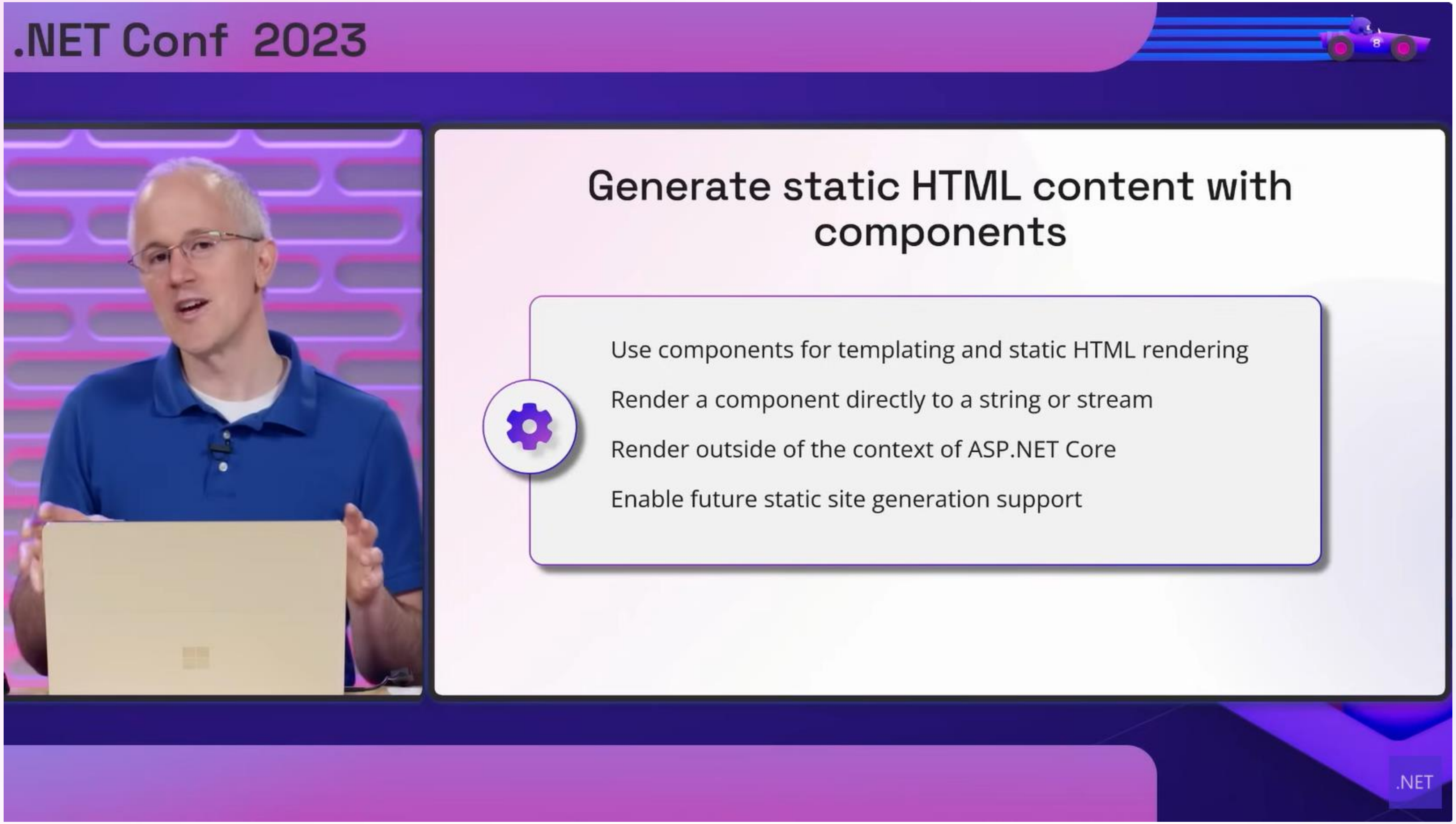
새로워진 Blazor



Additional Blazor enhancements in .NET 8

- QuickGrid
- Sections
- Routing
- Monitor circuit activity
- Improved authentication

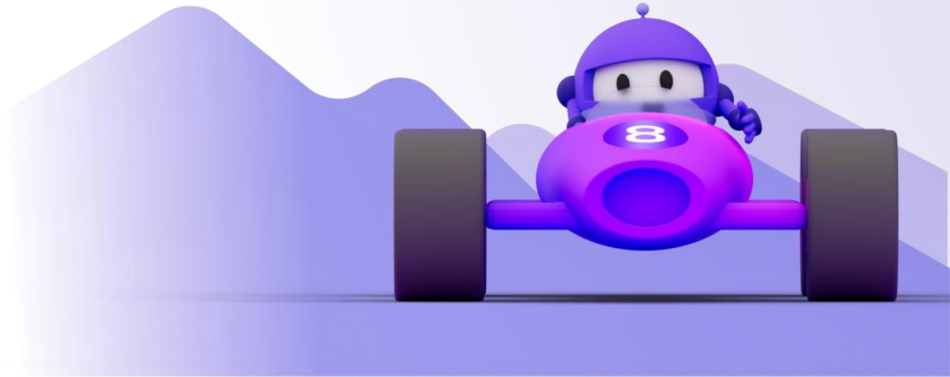
새로워진 Blazor



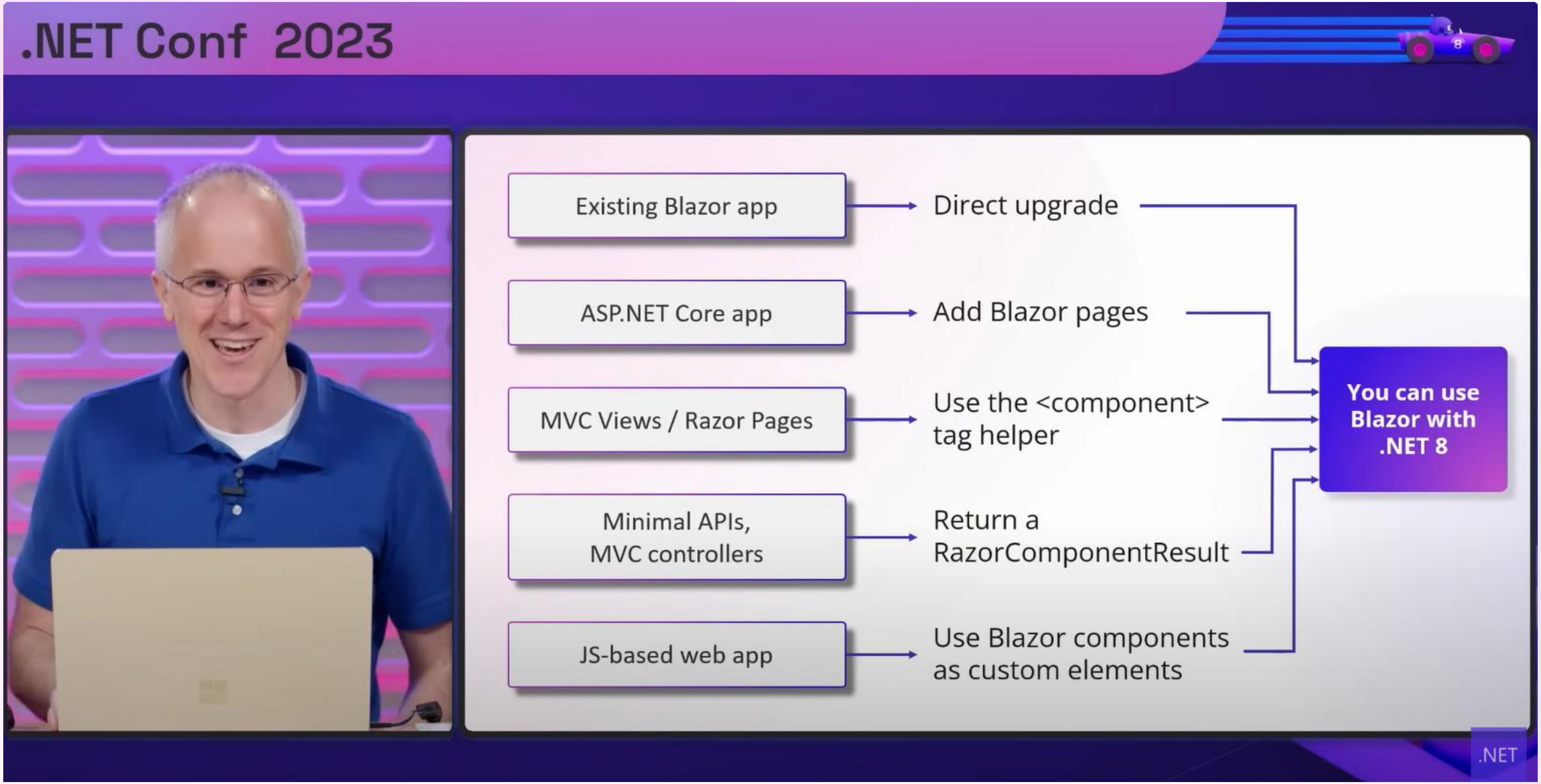
Generate static HTML content with components

- Use components for templating and static HTML rendering
- Render a component directly to a string or stream
- Render outside of the context of ASP.NET Core
- Enable future static site generation support

Demo



기존 프로젝트에서 Blazor 사용하기



.NET Conf 2023

Existing Blazor app → Direct upgrade

ASP.NET Core app → Add Blazor pages

MVC Views / Razor Pages → Use the <component> tag helper

Minimal APIs, MVC controllers → Return a RazorComponentResult

JS-based web app → Use Blazor components as custom elements

You can use Blazor with .NET 8

Q&A

Blazor Korea User Group

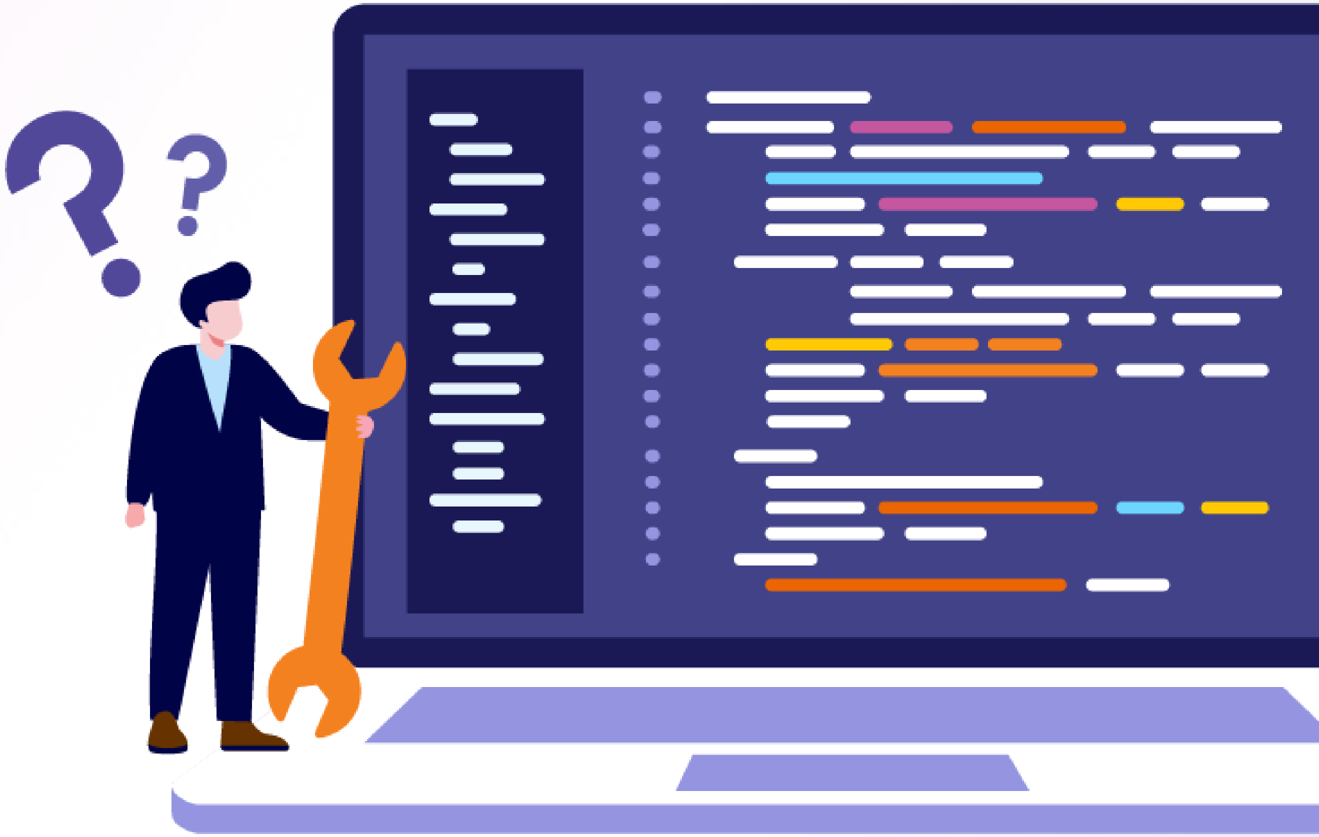
facebook.com/groups/blazor.korea

GitHub

https://github.com/blazorstudy

Home Page

https://blazorstudy.kr



THANK YOU