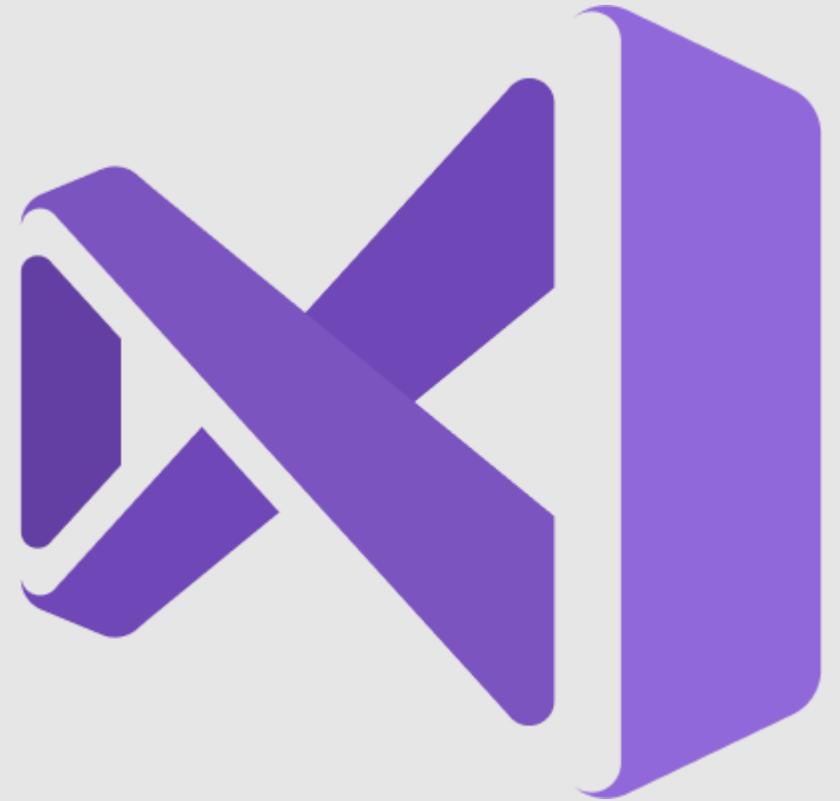


Web Apps with ASP.NET Core 3

Microsoft
Regional Director



Daniele Bochicchio
Chief Digital Officer @icubedsrl
Founder @aspitalia
Microsoft Regional Director, ASP.NET MVP
danieleb@aspitalia.com | @dbochicchio



Agenda

- Novità di ASP.NET Core 3
- Nuovi template e breaking changes
- EndPoint routing
- Worker e Generic Host
- JSON

Novità di ASP.NET Core 3

Microsoft.AspNetCore.App verrà snellito

Niente JSON.NET

Niente librerie esterne, come EF Core

Microsoft.AspNetCore.All è deprecato

Rimozione di API non più utili (ad es Session), marcate come Obsolete in precedenza

Solo per .NET Core

Rimosso supporto a .NET Framework

Novità di ASP.NET Core 3

C# 8 in Razor

SignalR: auto-reconnect e streaming, client C++

Performance

Authentication: Certificate, Kerberos, IdentityServer

Nuovi template

Bootstrap 4 come default per i template

Worker Service

Windows Service o applicazione in background

gRPC

Nuovo RPC moderno, open source e ad alte prestazioni

Blazor e Razor Component

Per Blazor

VS 2019 16.1, con .NET Core abilitato

Endpoint routing

ASP.NET Core 2.2 ha introdotto il concetto di Endpoint routing

Necessario per rendere più rapida la fase di determinazione dell'endpoint finale

ASP.NET Core 3 disaccoppia questa feature dal motore di MVC e la rende disponibile ovunque

```
app.UseRouting();  
...  
app.UseEndpoints(endpoints =>  
{  
    endpoints.MapGrpcService<GreeterService>();  
  
    endpoints.MapControllerRoute(  
        name: "default",  
        pattern:  
        "{controller=Home}/{action=Index}/{id?}");  
    endpoints.MapRazorPages();  
});
```

Async by default

Se una action finisce con il suffisso *Async*, questa non influenzerà il routing (in cui sarà omesso)

Integrazione con i nuovi async enumerable (*IAsyncEnumerable*)

Worker Service

Normali applicazioni console, già pronte con i servizi più diffusi
DI, Logging, configurazione, etc

Pronto per le diverse esigenze di far girare codice in background
Windows Service, Azure WebJob, Linux systemd, container

Nuovo generic host utilizzato anche dalle app ASP.NET

Può avere o non avere endpoint

Demo

Nuovi template Web e Generic Host



gRPC

Protocollo RPC open source e ad alte performance (inizialmente sviluppato da Google)

Disponibile su molte piattaforme/tecnologie/linguaggi

Contract-based

Formato binario basato su HTTP/2

Generalmente utilizzato nell'ambito dei servizi (*service-to-service*)

In architetture a micro-services, che senso ha utilizzare JSON (o un qualsiasi formato testuale) quando utilizzare una serializzazione binaria e non *human-readable* darebbe solo vantaggi?

Demo

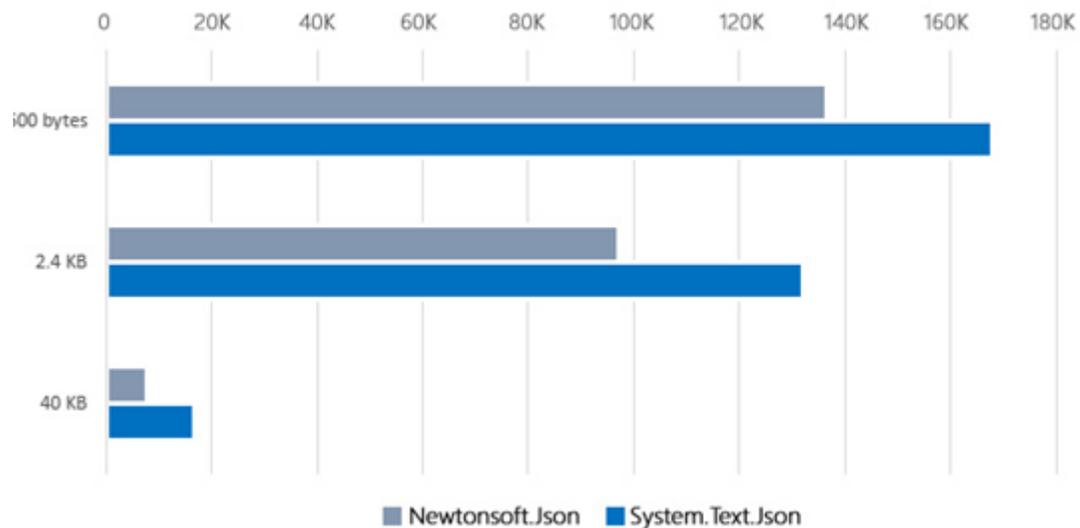
gRPC



JSON

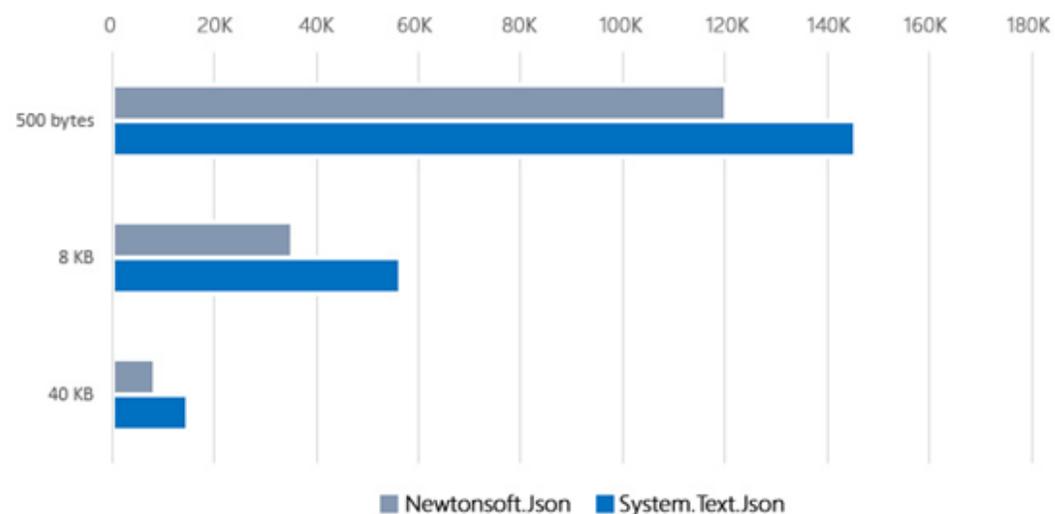
Deserializzazione JSON

Numero di operazioni al secondo nel deserializzare documenti JSON di 500 bytes, 2.4KB e 40KB.
(più è meglio)



Serializzazione JSON

Numero di operazioni al secondo nel serializzare documenti JSON di 500 bytes, 8KB e 40KB.
(più è meglio)



JSON

Nuovo namespace System.Text.Json

ASP.NET Core ha una storica dipendenza da JSON.NET

Sfrutta le migliorie introdotte da Memory<T>, Span<T>, etc

Classi ad hoc

JsonSerializer

JsonDocument

Utf8JsonReader e Utf8JsonWriter

(per grosse quantità di dati, in maniera efficiente)

Demo

JSON



ASP.NET Core 3

Ancora in preview, ma promette bene

Migliorie in chiave performance e nuovi trend

RTM a settembre

@dbochicchio
daniele@aspitalia.com

Slide e materiale su
<https://aspit.co/VS2019-live>

