



Microsoft
Regional Director

Developing modern web apps with ASP.NET Core

Daniele Bochicchio

Chief Digital Officer @icubed

Microsoft Regional Director, ASP.NET MVP

daniele@aspitalia.com | @dbochicchio



#netconfit



Agenda

.NET Core & ASP.NET Core Recap

Novità varie

Performance e...

Non solo ASP.NET Core



Cos'è .NET Core

Piattaforma di sviluppo

Cross-platform

Windows, macOS, Linux

Flexible deployment

Side-by-side o machine-wide

Open source su GitHub

Supportato da Microsoft

Eredita Base Class Library di .NET Framework 4.x

Rivisto: no dipendenze Windows

Compatibile via .NET Standard con codice per .NET Framework, Xamarin e Mono

Web == HTTP

Il web è diventato il *mezzo* principale su cui girano le applicazioni

Ormai troviamo endpoint web ovunque

Router, device di rete

IoT

TV, decoder, etc

Il web è diventato il nuovo modo attraverso il quale i device comunicano

HTTP+JSON = Protocollo universale di comunicazione

Nei prossimi anni questo fenomeno si accentuerà sempre di più

ASP.NET Core Recap: View to the max

TagHelper

Controlli, ciao-HtmlHelper

ViewComponent

Mini-Controller-View

Controller e Middleware

Se siete sviluppatori ASP.NET MVC, il passaggio è semplice

Se non lo siete ancora, non è mai troppo tardi per diventarlo!

Novità di ASP.NET Core 2

- Performance migliorate
 - Partenza più rapida, maggior numero di richieste al secondo
- Sviluppo immediato
 - Template semplificati
- Benefici introdotti da .NET Core 2
 - .NET Standard 2, performance, etc
- Refactoring di componenti chiave
 - Runtime, hosting con IIS, autenticazione, logging

Demo

Nuovo tipo di progetto e startup, view con MVC Core



.NET Core 2.1: prestazioni in compilazione

Prestazioni migliorate grazie alla compilazione

Build incrementali

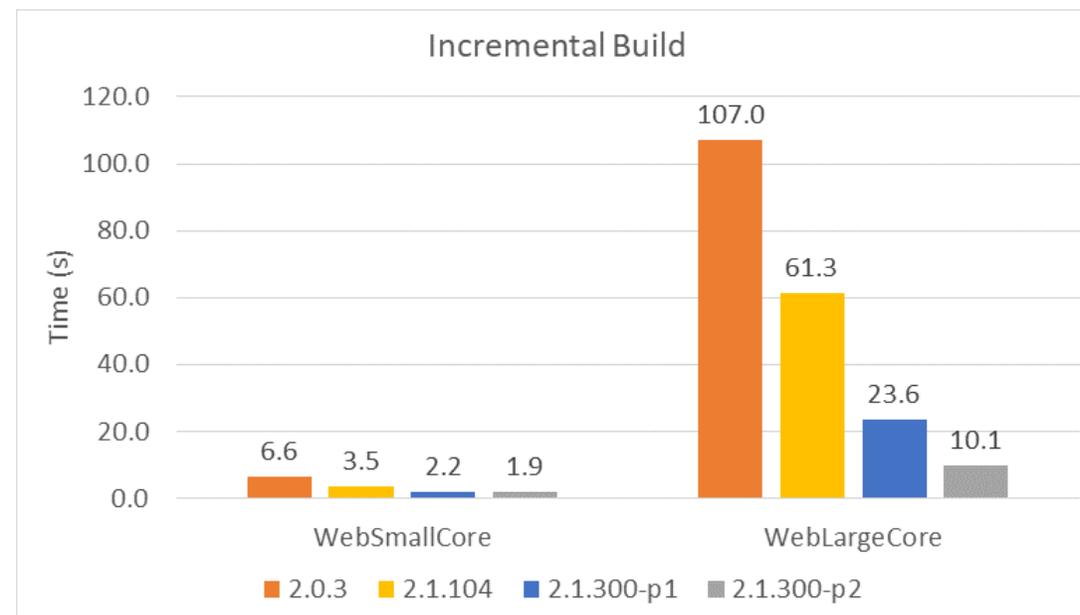
Il compilatore capisce quali parti sono effettivamente cambiate

- Speed up package asset resolution

- Speed up incremental package asset resolution

- MSBuild Node Reuse

- MSBuild ResolveAssemblyReferences cache



.NET Core 2.1

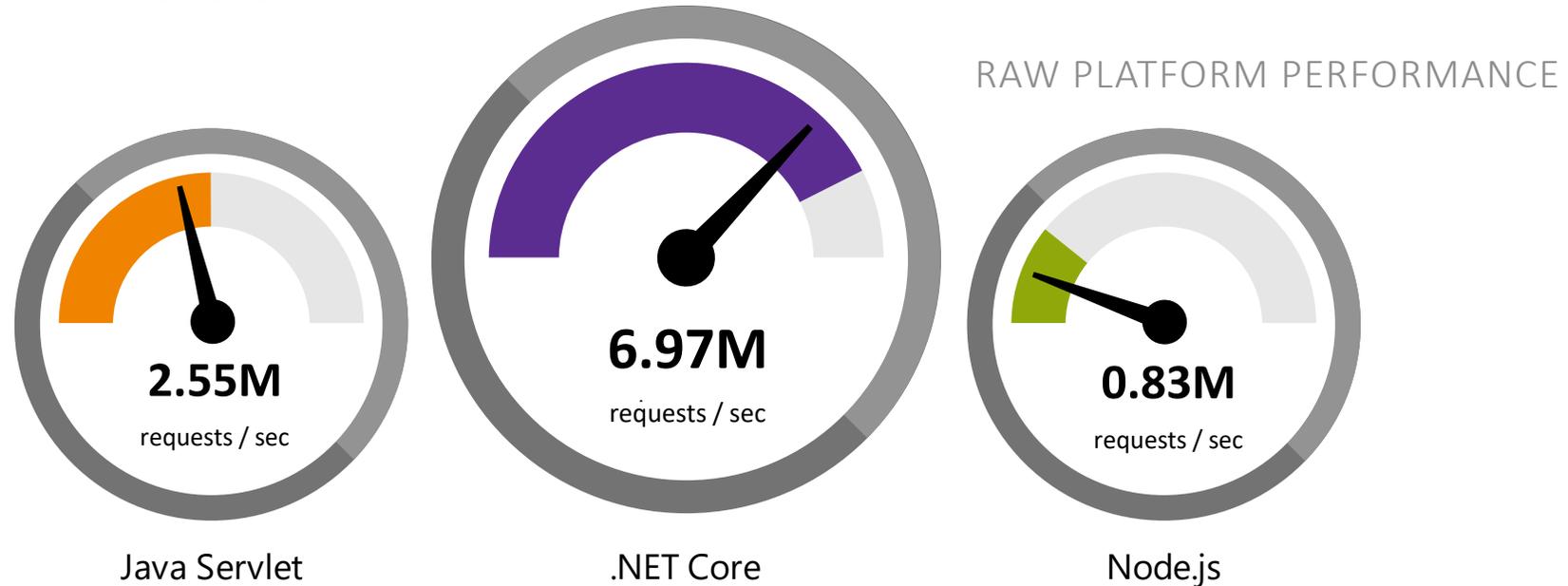
Supporto a ARM32 (Raspberry) anche per l'SDK

Prima era un community effort, ora è **supportato ufficialmente**

SDK disponibile anche su ARM32

- Ma la compilazione è veramente lenta

.NET == Fast

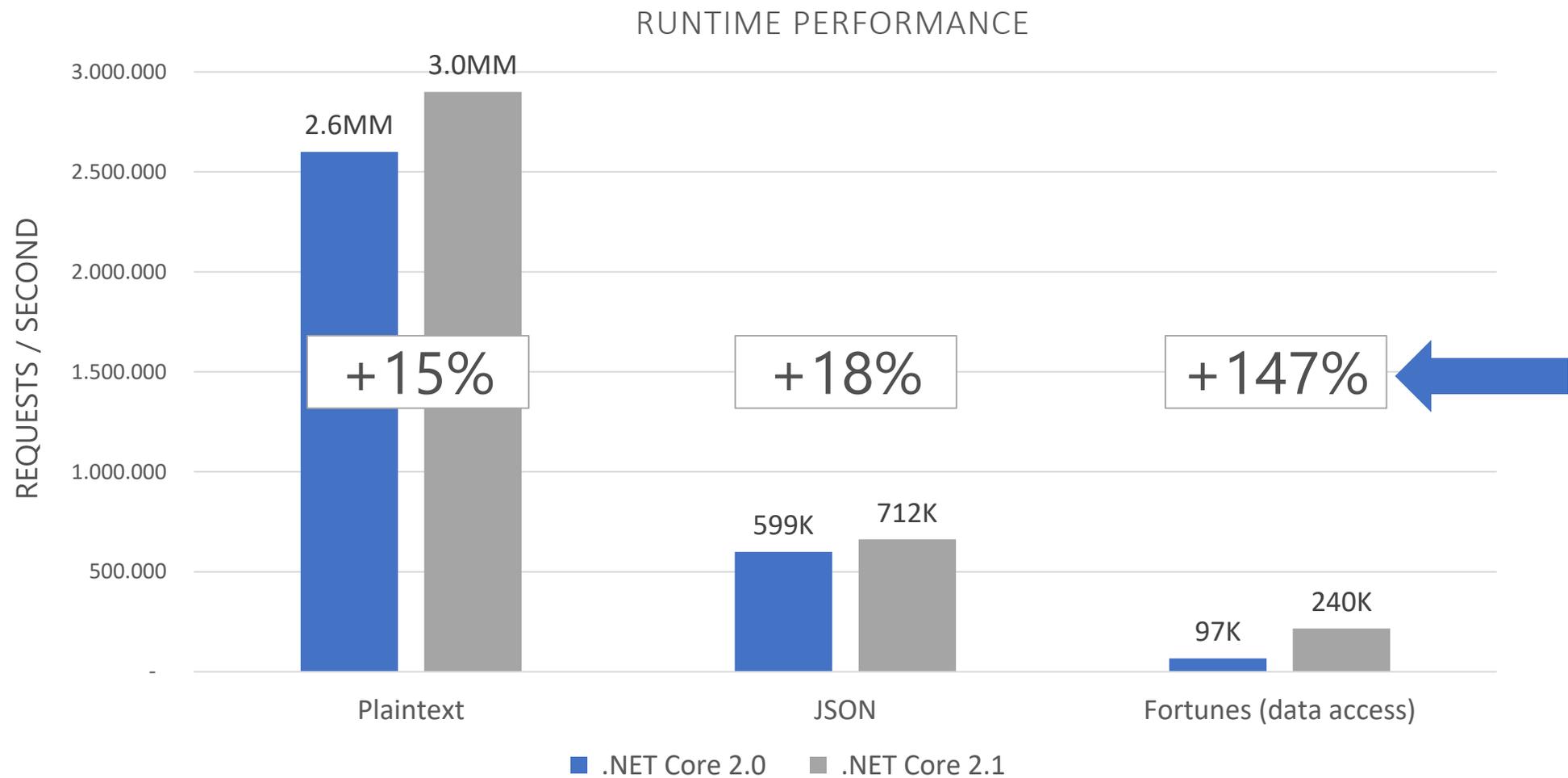


Info su [TechEmpower Round 16](#).

“Using the same-size server, we were able to go from 1,000 requests per second per node with Node.js to 20,000 requests per second with .NET Core.” — *Raygun*

microsoft.com/net/customers

.NET Core 2.1 su TechEmpower



Info su [TechEmpower round 16](#).

.NET Core 2.1

.NET Core

- Global Tools
- Span<T>
- Sockets
- HttpClient Performance
- Windows Compatibility Pack

EF Core

- Lazy Loading
- Value conversions
- Query types
- Data seeding

ASP.NET Core

- HTTPS
- Razor UI Library
- HttpClientFactory
- ASP.NET Core SignalR

Novità di ASP.NET Core 2.1

SignalR

Razor class
libraries

Identity UI library
& scaffolding

HTTPS

GDPR

MVC functional
testing

[ApiController],
ActionResult<T>

IHttpClientFactory

Kestrel on Sockets

Generic host
builder

Performance

Updated SPA
templates

.NET Core 2.2

.NET Core

- Multi-tier JIT compilation
- SQL Connection token auth

EF Core

- Provider per Cosmos DB
- Spatial extensions for SQL Server e SQLite
- Reverse engineering delle view di un database
- Collections di owned entities
- Query tagging

ASP.NET Core

- Template aggiornati: Bootstrap 4, Angular 6
- Migliorie a Web API, API security
- HTTP/2
- IIS in-process hosting
- Health checks
- Endpoint routing
- SignalR Java client

Demo



Non si vive di solo server

Ormai lo sviluppo web abbraccia anche altre cose

Docker

DevOps

Progressive Web Apps

Docker e ASP.NET Core 2.1

Immagini basate su Alpine (distro Linux ridotta all'osso)

- 2.1-aspnetcore-runtime-alpine
 - 2.1-runtime-deps-alpine*
-
- * Solo dependencies, senza runtime, per self-contained deployment

Solo 83 MB anziché 219 come baseline

- <https://hub.docker.com/r/microsoft/dotnet-nightly/>

Progressive Web Apps

PWA = applicazioni web che diventano app

Si possono scrivere con i framework preferiti

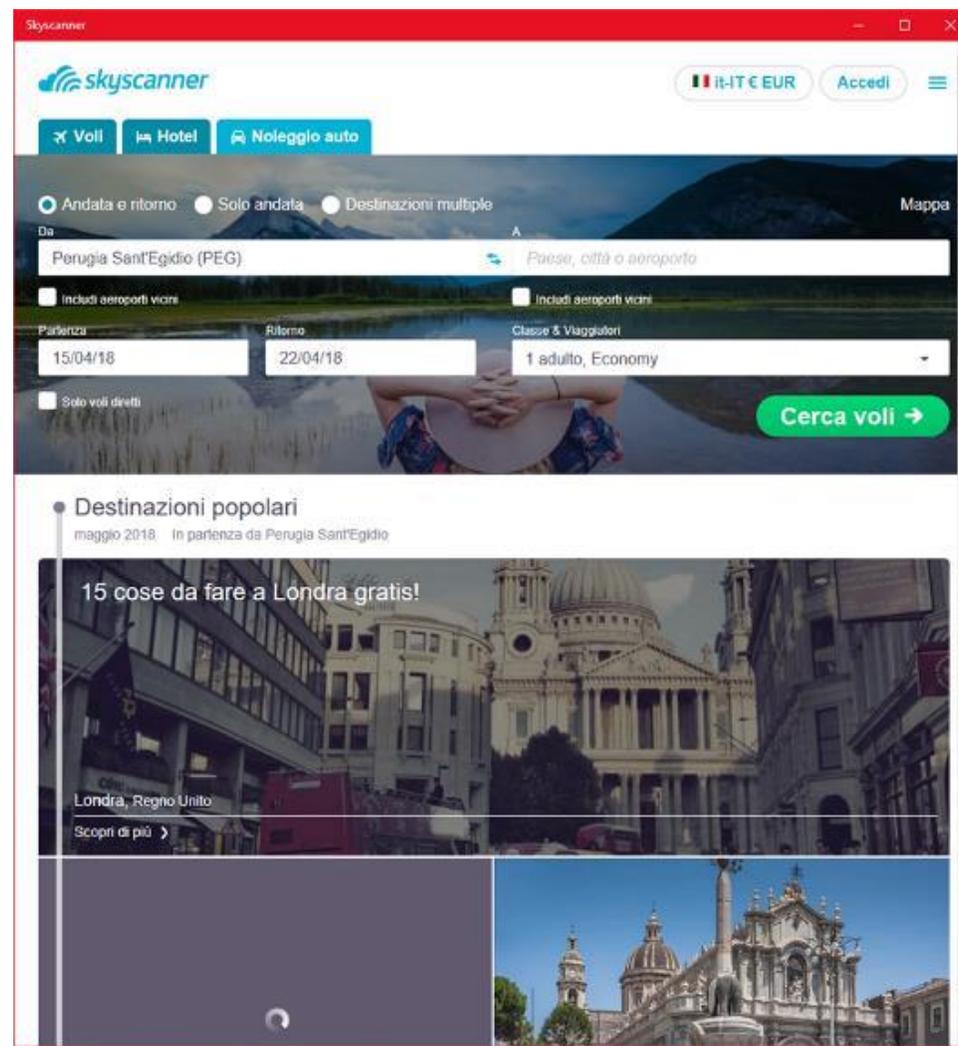
- Angular, React, vue.js

Le PWA devono

- essere responsive e fluide
- funzionare offline

ASP.NET Core può essere il collante

Servizi, generazione view statiche, etc



Progressive Web Apps

Tecnicamente, una PWA sfrutta questi concetti

- **Manifest:** descrive l'app e i suoi contenuti
- **Service Worker:**
 - consente di eseguire processi in background, ricevere notifiche push
 - gestisce la cache
 - gestisce le chiamate di rete
- In futuro, grazie a **WebAssembly**, diventerà possibile avere PWA che sono state scritte in un qualsiasi linguaggio e compilate in WASM, un formato portabile (binario) per VM, con codice eseguito direttamente nel browser
- Sono **applicazioni native** a tutti gli effetti

Demo



Accattatevillo! (cit)

Nuovo libro su ASP.NET Core 2(.1)
uscito a settembre con Hoepli

<http://books.aspitalia.com/ASP.NET-Core/>



Grazie!

@dbochicchio
daniele@aspitalia.com

Materiale su
<http://aspit.co/netconf-18>

