

# Upgrade your Existing Web APIs

To ASP.NET Core APIs

# Hi, I'm J.

Jonathan "J." Tower

Partner & Principal Consultant

Trailhead Technology Partners



🏆 Microsoft MVP in ASP.NET

🏆 Telerik/Progress Developer Expert

📅 Organizer of Beer City Code

✉ jtower@trailheadtechnology.com

🌐 trailheadtechnology.com/blog

🐦 jtowermi

[github.com/jonathantower/api-to-dotnet-core](https://github.com/jonathantower/api-to-dotnet-core)

# Preview

- Differences between ASP.NET Web API and ASP.NET Core APIs
- Migrating your Web API to ASP.NET Core
- Using the Compatibility Shim

# If You Give \$100, So Will I!

## **[bit.ly/techbash-water](https://bit.ly/techbash-water)**

*"charity:water is a non-profit organization that provides clean and safe drinking water to people in developing nations. The organization was founded in 2006 and has helped fund 22,936 projects in 24 countries, benefiting over 4.6 million people." - Wikipedia*

*"4/4 Stars"  
- CharityNavigator.org*

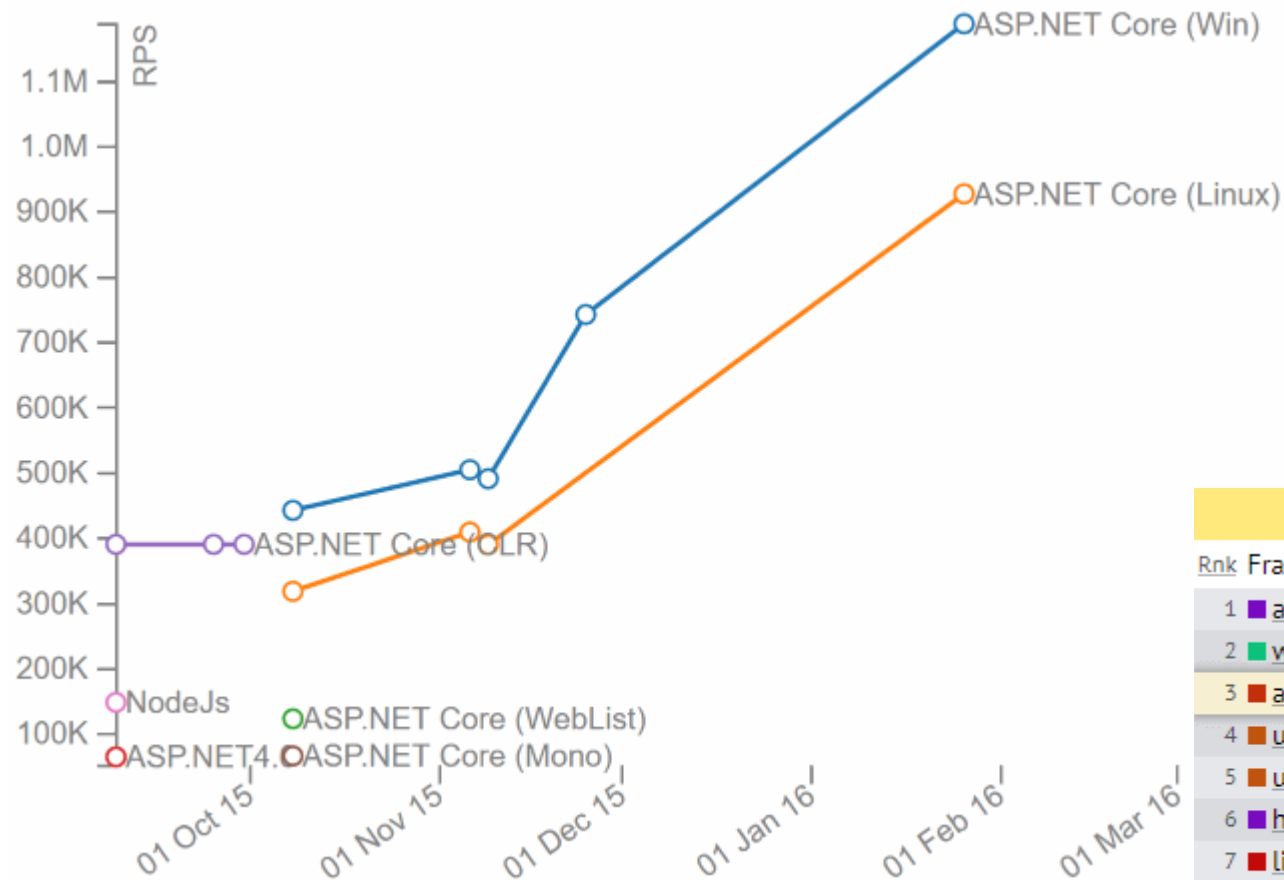


charity: water

# What's Different?

ASP.NET Web API vs ASP.NET Core API

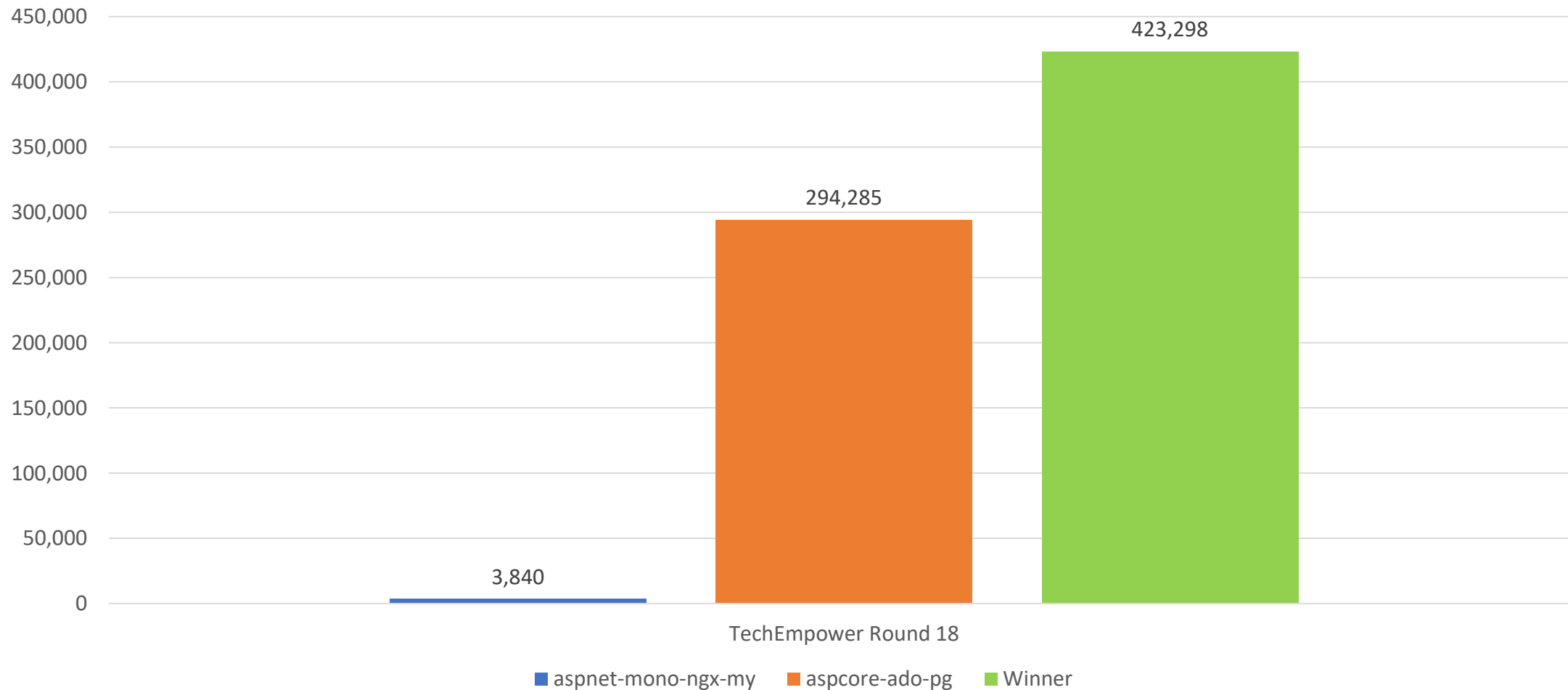
# Performance



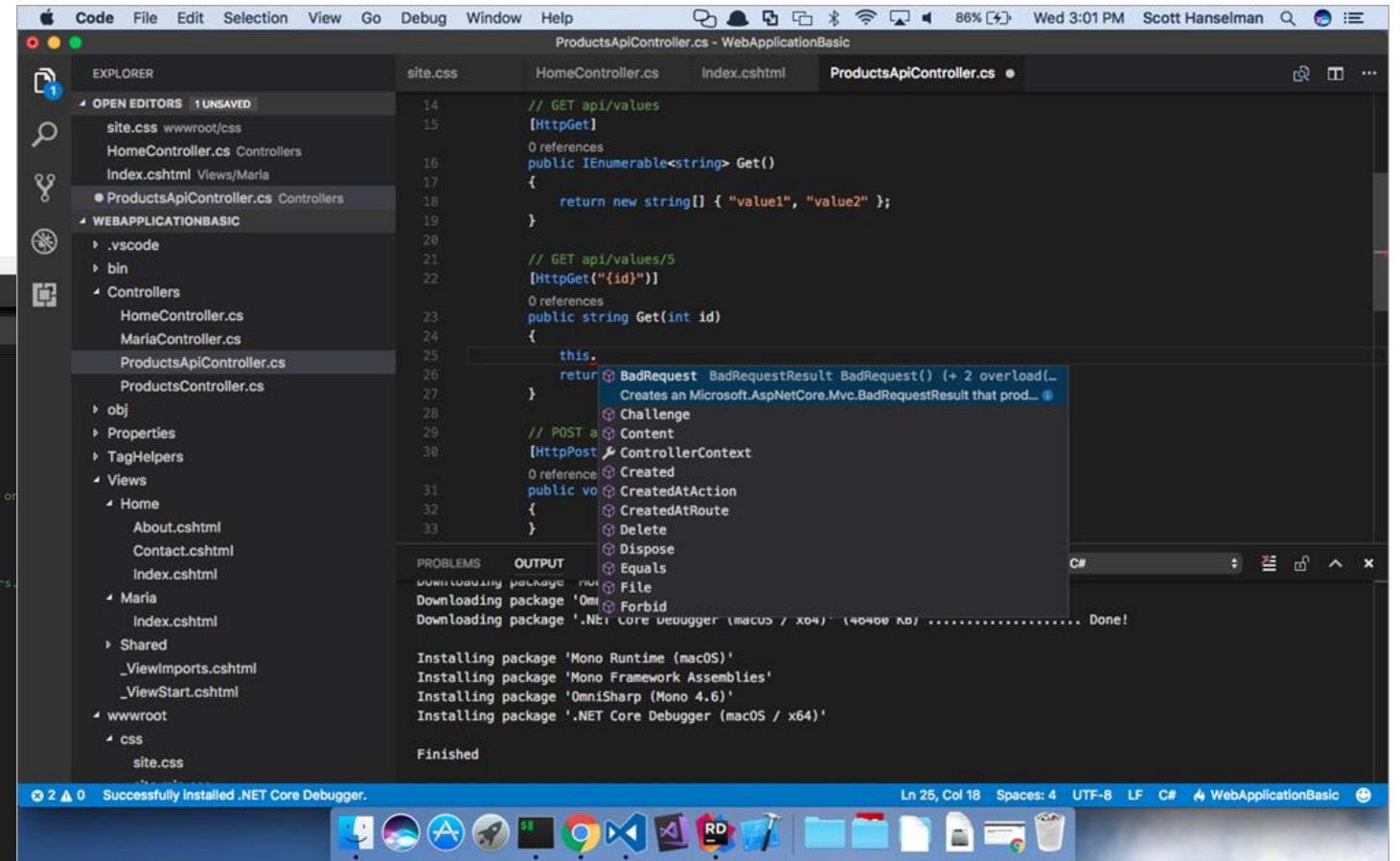
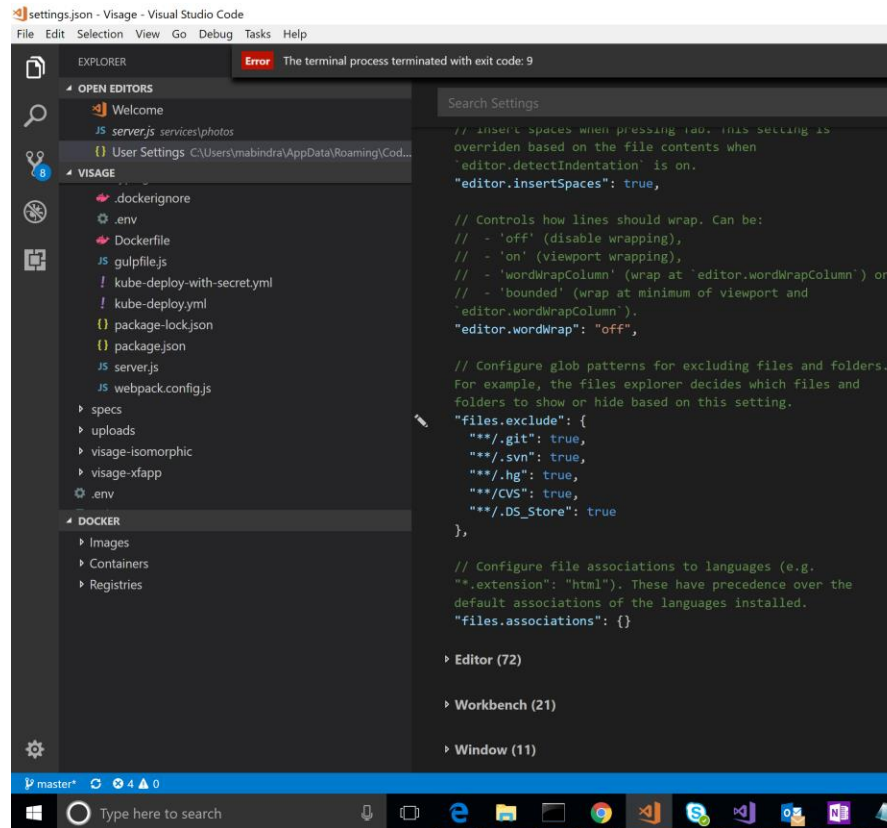
Best plaintext responses per second, Test environment (331 tests)

Rnk	Framework	Best performance (higher is better)	Cls	Lng	Plt	FE
1	actix-raw	7,003,320   100.0%	Plt	Rus	Non	act
2	wizzardo-http	7,002,116   100.0%	Mcr	Jav	Non	Non
3	aspcore	7,000,118   100.0%	Plt	C#	.NE	kes
4	ulib	6,998,365   99.9%	Plt	C++	Non	ULi
5	ulib-plaintext_fit	6,998,068   99.9%	Plt	C++	Non	ULi
6	hyper	6,996,874   99.9%	Mcr	Rus	Rus	Hyp
7	libreactor	6,996,726   99.9%	Mcr	C	Non	Non
8	tokio-minihttp	6,995,981   99.9%	Mcr	Rus	Rus	tok
9	baseio-http-lite	6,983,540   99.7%	Plt	Jav	bas	Non
10	aspcore-rhtx	6,975,733   99.6%	Plt	C#	.NE	kes

# TechEmpower Round 17

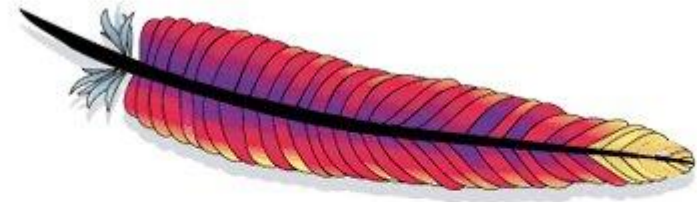


# Cross-Platform Development

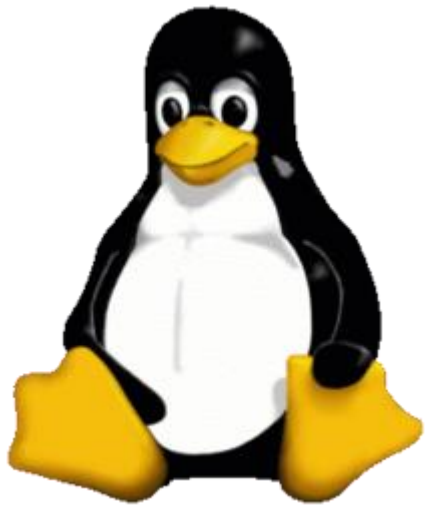




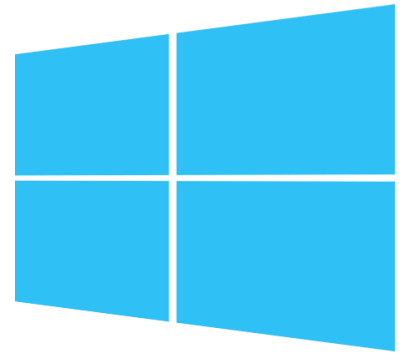
# Cross-Platform Hosting



# Apache



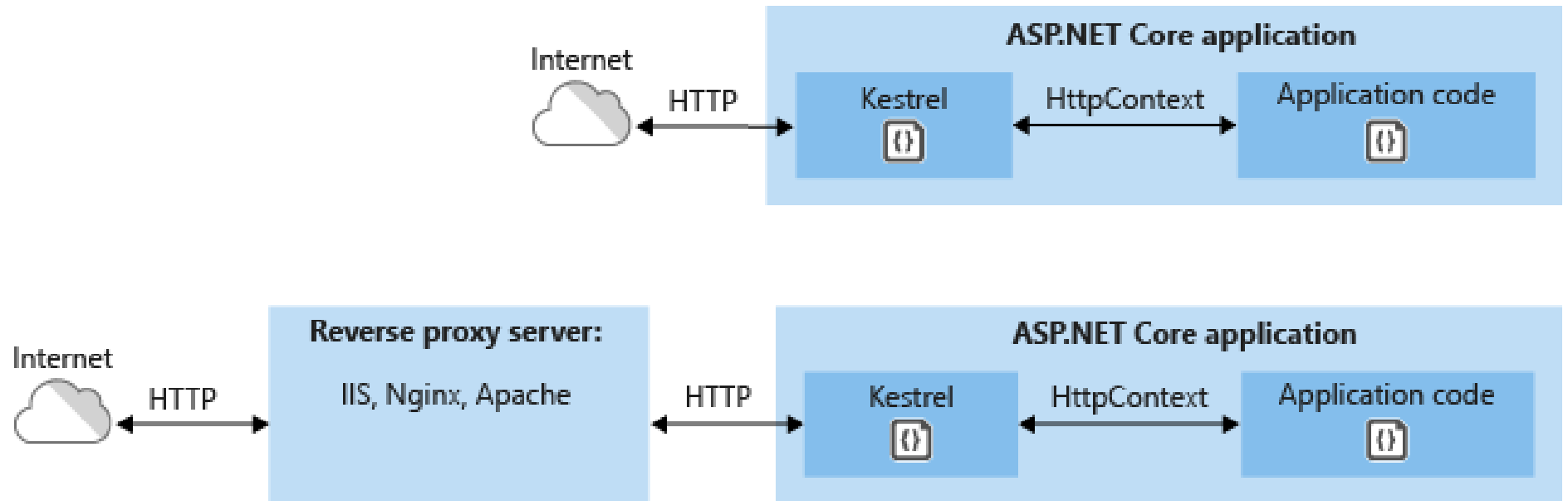
Mac OS



# NGINX

# Kestrel

- Decouples ASP.NET Core & IIS
- Cross-platform
- Built for speed
- Limited features (port sharing)



# Side-by-side Versioning



- Framework-dependent vs self-contained
- No worries about missing features
- Deploy without fear of breaking old apps

# Configuration

- JSON based
- Also supports XML, Azure key valut, command line, environment variables, environment files
- Easier hierarchical support
- IOptions<T>





# Logging

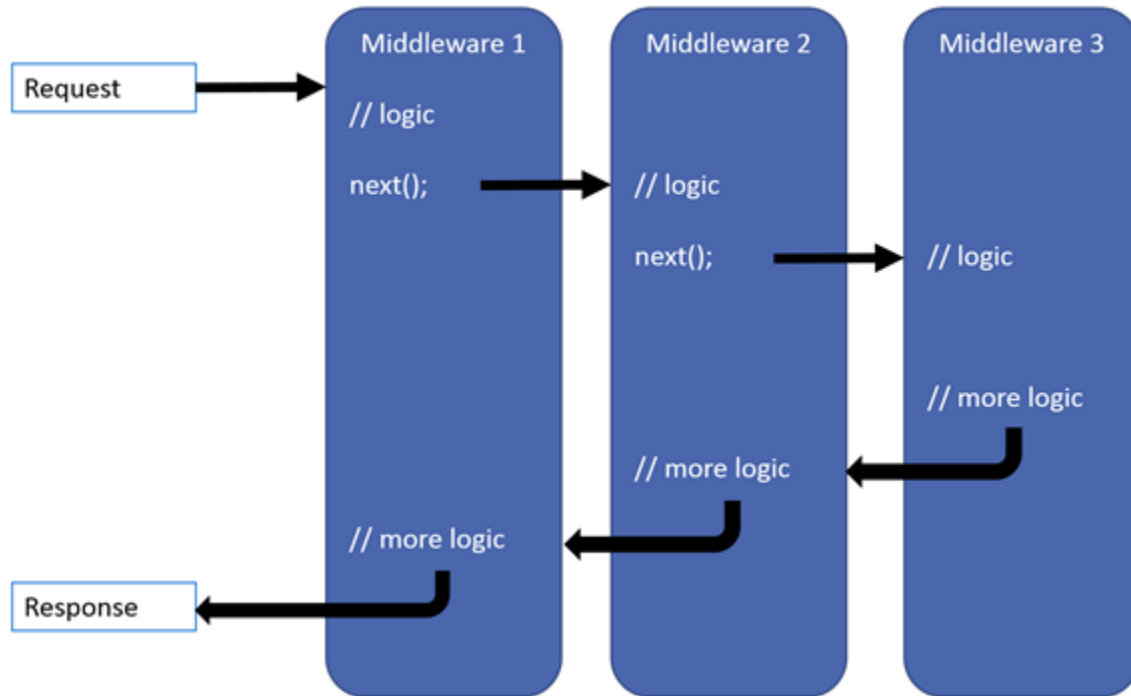


- Built-in and 3<sup>rd</sup> party providers
  - Console, Debug, EventSource, EventLog, TraceSource
  - AzureAppServices, ApplicationInsights
  - elmah, Log4net, NLog, Serilog, others...

# Startup Configuration

- Startup Class
  - Configure() - middleware/pipeline
  - ConfigureServices() - services/DI

# Middleware



- A little like HTTPModules
- Configured in code (not XML)
- Easy control over the order

**DEMO: Timing middleware**

# Dependency Injection

- Built-In
- Used by the framework itself
- Simple lifetime options:
  - Singleton
  - Scoped (HTTP request)
  - Transient
- Easily integrate third-party DI frameworks





# Controller Base Classes

## **ASP.NET Web API**

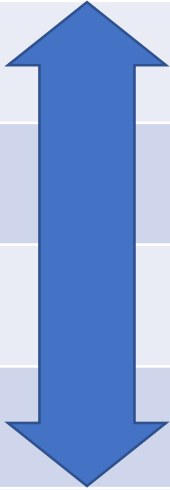
- `System.Web.Mvc.Controller`
- `System.Web.Http.ApiController`

## **ASP.NET Core**

- `Microsoft.AspNetCore.Mvc.Controller`
- `Microsoft.AspNetCore.Mvc.ApiControllerAttribute`



# .NET Standard

Class Library that targets	APIs	Runs on .NET Core	And on .NET Framework
.NET Standard 1.0	Fewer	1.0 and up	4.5 and up
.NET Standard 1.2		1.0 and up	4.5.1 and up
.NET Standard 1.3		1.0 and up	4.6 and up
.NET Standard 1.6		1.0 and up	4.6.1 and up
.NET Standard 2.0		2.0 and up	4.6.1 and up
.NET Standard 2.1	More	3.0 and up	N/A *

\* .NET Framework will not support .NET Standard 2.1 *or any other later version*.

# Migration Steps

ASP.NET Web API to ASP.NET Core API

# Special Thanks To **Scott Addie & Steve Smith**

Microsoft Docs: Migrate from ASP.NET Web API to ASP.NET Core

<https://docs.microsoft.com/en-us/aspnet/core/migration/webapi>

# Source Project (ASP.NET Web API 2)

Get Started with ASP.NET Web API 2

<https://docs.microsoft.com/en-us/aspnet/web-api/overview/getting-started-with-aspnet-web-api/tutorial-your-first-web-api>

# Caveats

- No two projects are the same; same steps won't be enough for **all projects**
- Your project is probably **more complicated** than the sample
- Your project probably includes **references** to other projects that need to be updated

# Step 1 // Review the Source Project

- Review Global.asax
- Review App\_Start folder
- Review ProductsController
- Look at nuget and project references
- Look at configuration settings

## Step 2 // New ASP.NET Core Project

- File > New > Project > ASP.NET Core Web Application
- Target .NET Core 2.1
- API project template
- Delete unused code files



# Step 3 // Migrate Configuration

- Startup
  - `app.UseMvc()`
    - includes attribute routing by default
    - includes default MVC route
- Inject configuration as needed

# Step 4 // Migrate Models & Controllers

- Copy controller and model classes
  - Fix namespaces
  - Change ApiController -> Controller
  - Remove any System.Web.Http references
  - Change IHttpActionResult to IActionResult

# Step 5 // Configure Routing

- Configure controller attributes
  - [Route("api/[controller]"), ApiController]
- Configure action method attributes
  - [HttpGet] GetAllProducts()
  - [HttpGet("{id}")] GetProduct(id)

Let's Try It Out!

# Compatibility Shim

# Compatibility Shim\*

1. **ApiController** base class
2. Enables Web API-style **model binding** (works like ApiControllerAttribute)
3. Adds back **HttpRequestMessage**, **HttpResponseMessage**.
4. Adds back **CreateResponse<T>()** and **CreateErrorResponse()**
5. Adds back action result methods:
  - **BadRequestErrorMessageResult()**
  - **ExceptionResult()**
  - **InternalServerErrorResult()**
  - **InvalidModelStateResult()**
  - **NegotiatedContentResult()**
  - **ResponseMessageResult()**

\*Only ASP.NET CORE 2.1 and before

# Installing Compatibility Shim

1. Install **Microsoft.AspNetCore.Mvc.WebApiCompatShim** from NuGet
2. `services.AddMvc().AddWebApiConventions()` in `Startup.ConfigureServices`.
3. API-specific routes using **MapWebApiRoute** in `UseMvc()`

\*Only ASP.NET CORE 2.1 and before

# Recap

- Differences between ASP.NET Web API and ASP.NET Core APIs
- Migrating your Web API to ASP.NET Core
- Using the Compatibility Shim



# If You Give \$100, So Will I!

## **[bit.ly/techbash-water](https://bit.ly/techbash-water)**

*"charity:water is a non-profit organization that provides clean and safe drinking water to people in developing nations. The organization was founded in 2006 and has helped fund 22,936 projects in 24 countries, benefiting over 4.6 million people." - Wikipedia*

*"4/4 Stars"  
- CharityNavigator.org*



charity: water

# Thanks! Questions?

Jonathan "J." Tower

Partner & Principal Consultant

Trailhead Technology Partners



Microsoft MVP in ASP.NET



Telerik/Progress Developer Expert



Organizer of Beer City Code



**TRAILHEAD**  
TECHNOLOGY PARTNERS

[trailheadtechnology.com](http://trailheadtechnology.com)



[jtower@trailheadtechnology.com](mailto:jtower@trailheadtechnology.com)



[trailheadtechnology.com/blog](http://trailheadtechnology.com/blog)



[jtowermi](https://twitter.com/jtowermi)

[github.com/jonathantower/api-to-dotnet-core](https://github.com/jonathantower/api-to-dotnet-core)