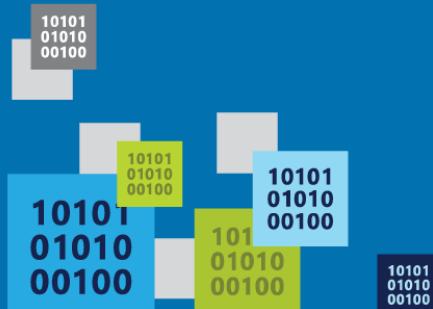


ASP.NET Core* in 2017

The Future of Web Apps

Mr. Xuan Tinh Chu

<http://bachkhoasoft.com>



CIT

Agenda

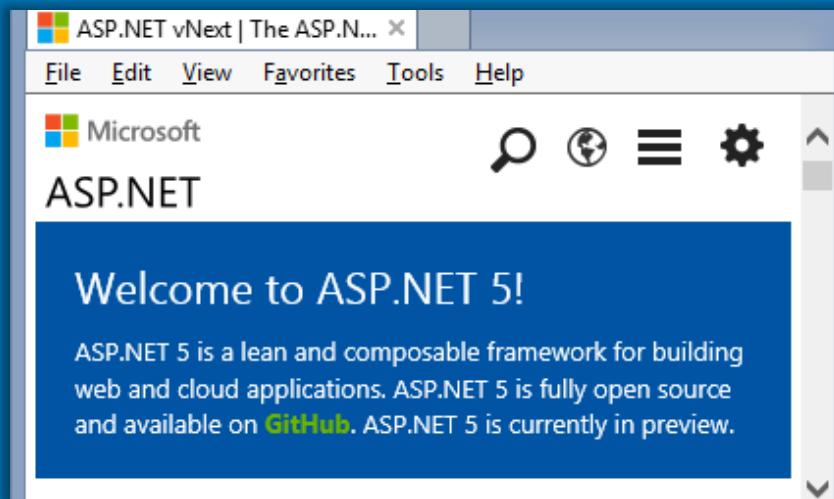
Introduction

- > .NET (Framework & Core)
- > ASP.NET Core
- > Visual Studio

Q&A

Introduction

Sources



http://blogs.msdn.com/b/webdev/archive/2015/02/23/aspnet-5-updates-for-feb-2015.aspx x

MSDN Blogs x

File Edit View Favorites Tools Help

ASP.NET 5 Updates and other improvements for Web Developers in Visual Studio 2015 CTP 6

Jeffrey T. Fritz 23 Feb 2015 7:00 AM 34 RATE THIS x

ASP.NET 5 Updates and other improvements for Web Developers in Visual Studio 2015 CTP 6

Jeffrey T. Fritz 23 Feb 2015 7:00 AM 34 RATE THIS x

MSDN Blogs, Feb 2015 Update

asp.net/vNext x

ScottGu's Blog - Introducin... x

File Edit View Favorites Tools Help

ScottGu's Blog

Introducing ASP.NET 5

Monday, February 23, 2015

.NET ASP.NET Community News

Visual Studio 2017 Laun... x

https://channel9.msdn.com/E... x

Channel 9 x

Visual Studio 2017 Launch

Channel 9 BROWSE FORUMS CODING4FUN EVENTS

Follow Us @Ch9

dotnetConf 2015 March 18 Save the date!

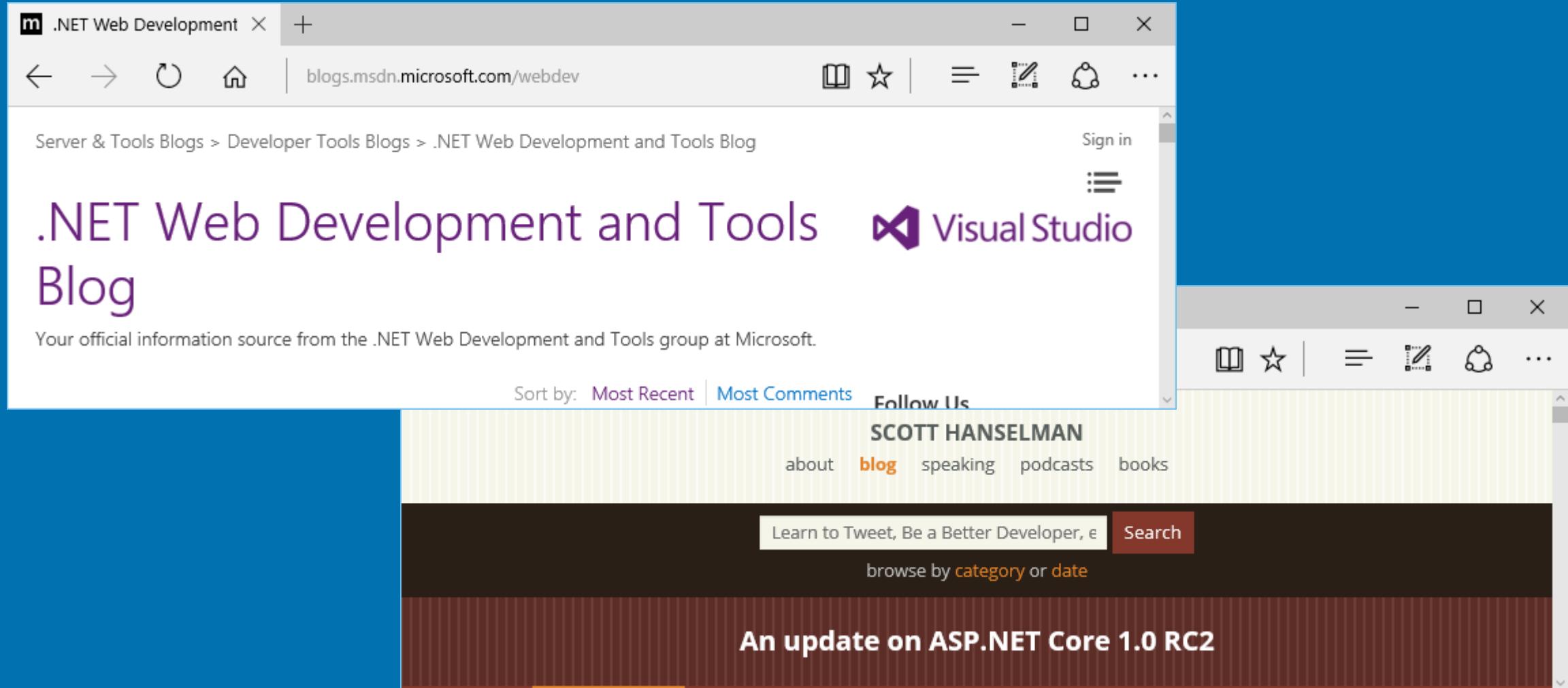
Scott Guthrie's Blog

https://channel9.msdn.com

dotnetConf 2015 on MSDN Ch9

Blog Posts in 2016

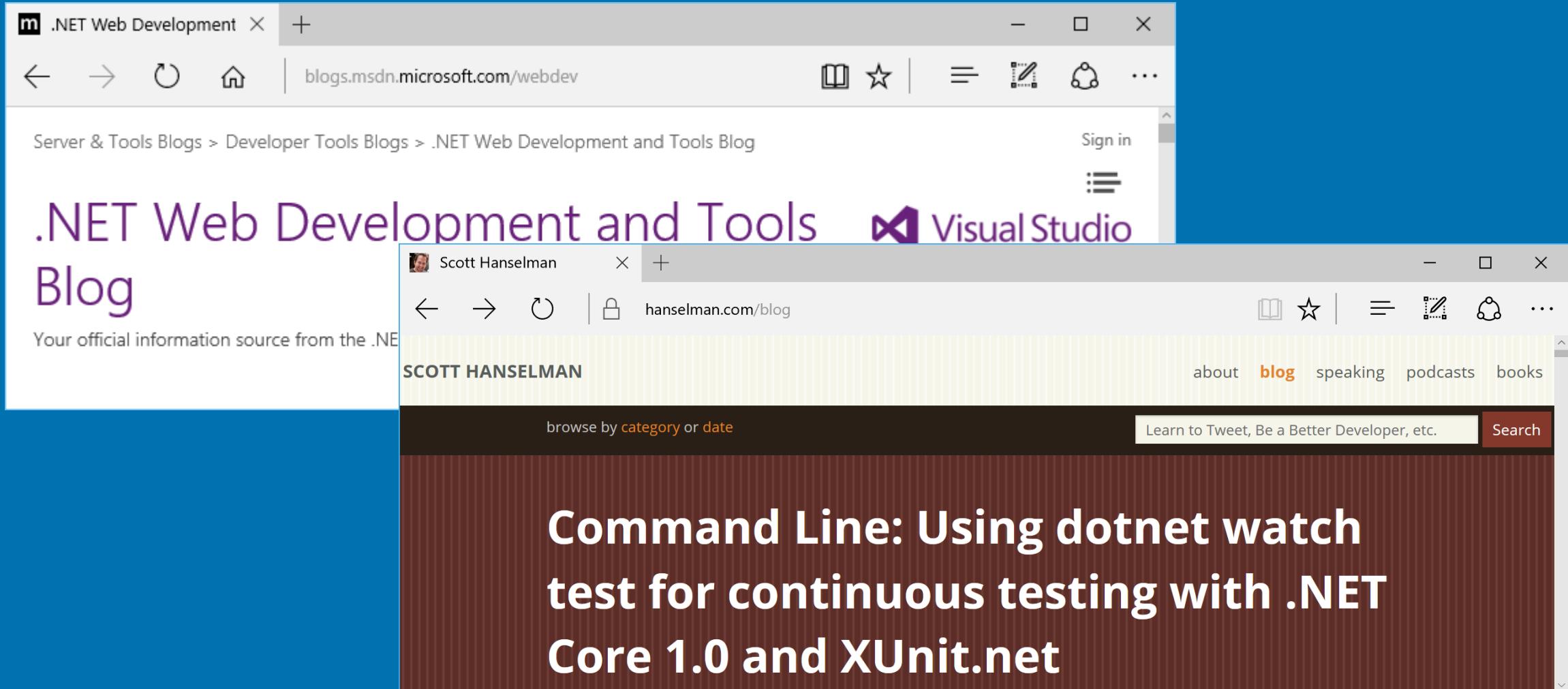
Community Standup Notes: <https://blogs.msdn.microsoft.com/webdev/>



RC2: <http://www.hanselman.com/blog/AnUpdateOnASPNETCore10RC2.aspx>

Blogs To Read in 2017

Community Standup Notes: <https://blogs.msdn.microsoft.com/webdev/>



Scott Hanselman's Blog: <https://www.hanselman.com/blog/>

Visual Studio 2017 Launch Videos

The screenshot shows a Microsoft Edge browser window displaying the 'Visual Studio 2017 Launch' channel on Channel 9 (msdn.com). The page title is 'Visual Studio 2017 Launch'. The main content area features a purple banner with the Microsoft logo and 'Visual Studio 2017 Launch' text, indicating there are 69 episodes. Below the banner, a summary text states: 'Mar 07–8, 2017. We're excited to announce that Visual Studio 2017 will be available for download on Tuesday, March 7, 2017! Join us online for a two-day virtual event celebrating 20 years of Visual Studio and our latest release.' It also mentions that on March 7, live streams will feature Julia Liuson, Brian Harry, Miguel de Icaza, and Scott Hanselman sharing innovations in Visual Studio, .NET, Xamarin, Azure, and more. A 'show more' button is present. Below this, a section titled '63 Speakers:' shows small profile pictures of various speakers. At the bottom, a 'Most Viewed' section displays three video thumbnails: 'Visual Studio 2017 Launch Keynote' (02:06:16), 'What's new for .NET Core and Visual Studio 2017?' (08:20), and 'Exploring C# Productivity in Visual Studio 2017' (29:58).

<https://channel9.msdn.com/Events/Visual-Studio/Visual-Studio-2017-Launch?sort=viewed&direction=asc>

Build Conference

<http://build.microsoft.com>



Docs + Tutorials

Docs: <https://blogs.msdn.microsoft.com/webdev/2017/02/07/asp-net-documentation-now-on-docs-microsoft-com/>

The image displays two separate browser windows side-by-side, illustrating the transition from MSDN Blogs to the new documentation platform.

Top Browser Window: This window shows a blog post titled "ASP.NET Documentation Now on docs.microsoft.com". The URL in the address bar is blogs.msdn.microsoft.com/webdev/2017/02/07/asp-net-documentation-now-on-docs-microsoft-com/. The post is dated February 7, 2017, by Jeffrey T. Fritz, with 11 comments. It includes social sharing links for Facebook (288 shares), Twitter (136 tweets), and LinkedIn (34 shares).

Bottom Browser Window: This window shows a tutorial titled "Building your first ASP.NET Core MVC app with Visual Studio". The URL in the address bar is docs.microsoft.com/en-us/aspnet/core/tutorials/first-mvc-app. The page features a navigation sidebar with "Introduction", "Getting Started" (which is expanded to show sub-sections like "What's New in ASP.NET Core", "ASP.NET Core 1.0", "ASP.NET Core 1.1", and "ASP.NET Core 2.0"), and a "Download PDF" link. The main content area describes the tutorial series: "This series of tutorials will teach you the basics of building an ASP.NET Core MVC web app using Visual Studio." It also includes a "Contributors" section with four profile icons and a "Feedback" button.

Tutorials: <https://docs.microsoft.com/en-us/aspnet/core/tutorials/first-mvc-app/>

.NET Core for Cross-Platform Dev

Info and Downloads: <https://www.microsoft.com/net>

The screenshot shows a Microsoft Edge browser window displaying the official .NET website at [microsoft.com/net](https://www.microsoft.com/net). The page has a blue header with the Microsoft logo and the text ".NET - Powerful Open S...". Below the header, there's a navigation bar with back, forward, refresh, and search icons. The main content area features a large tagline "Any developer, any app, any platform". Three main sections are displayed below: "Windows development" (with a Windows logo icon), "Cross-platform server apps" (with a server rack icon), and "Mobile apps on any device" (with icons for Android, Windows, and iOS). Each section includes a brief description and a "Read more" link.

Windows development

The .NET framework helps you create mobile, desktop, and web applications that run on Windows PCs, devices and servers and it's included in Visual Studio.

Cross-platform server apps

Develop web sites and services that run on Linux, Windows and macOS with the blazing fast and modular platform provided by .NET Core and ASP.NET Core.

Mobile apps on any device

Xamarin brings the power and productivity of .NET to iOS and Android, reusing skills and code while getting access to the native APIs and performance.

ASP.NET Core 1.0

Info and Downloads: <http://www.asp.net/>

The screenshot shows the ASP.NET Core 1.0 website homepage. The header includes the Microsoft logo and navigation links for Home, Get Started, Learn, Hosting, Downloads, Community, Forums, and Help. A search bar and language selection are also present. The main content features a large "Get building" heading with an illustration of people working on laptops and a globe icon. Below this is a section titled "ASP.NET" with a description of the framework as an open-source web framework for building modern web apps and services with .NET. It includes download links for "Visual Studio 2017" and ".NET Core". The "Free courses" section on the left offers learning resources. A red box highlights the "Announcements" section on the right, which lists several recent updates.

Register for the ASP.NET Core sessions presented by Scott Hunter, Jeff Fritz, and Dan Roth from the .NET/ASP.NET team at DEVintersection 2017! [Learn more >](#)

Microsoft | [Blog](#) | [...](#)

Search [Language](#) [...](#)

ASP.NET [Home](#) [Get Started](#) [Learn](#) [Hosting](#) [Downloads](#) [Community](#) [Forums](#) [Help](#)

Get building



ASP.NET

ASP.NET is an open source web framework for building modern web apps and services with .NET. ASP.NET creates websites based on HTML5, CSS, and JavaScript that are simple, fast, and can scale to millions of users.

[Download Visual Studio 2017](#)
Free, powerful IDE for ASP.NET on Windows

[Download .NET Core](#)
Free .NET command-line tools for Windows, Mac, and Linux

Free courses



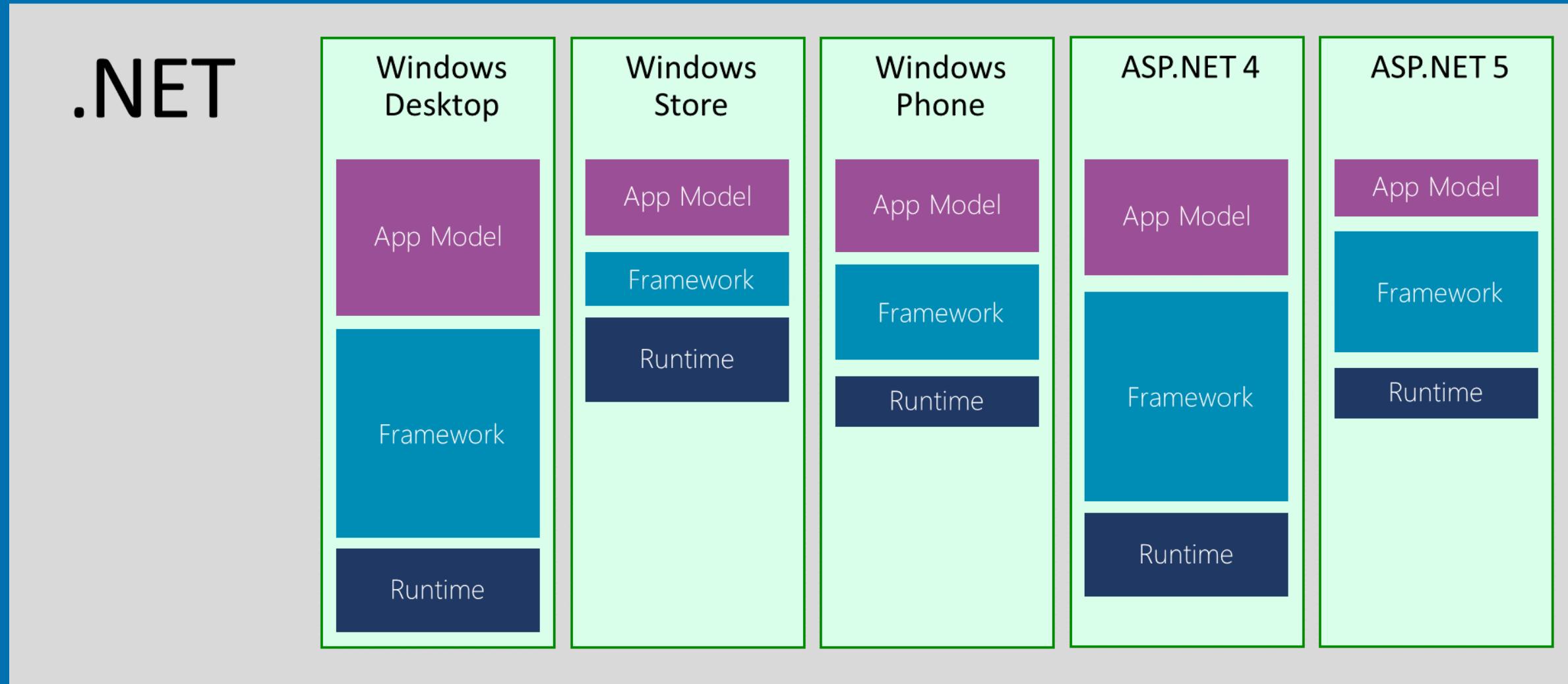
Start learning ASP.NET and ASP.NET Core for free with videos, interactive tutorials, code samples and more.

[Learn more >](#)

Announcements

Visual Studio 2017 Launch: March 7-8
ASP.NET Documentation Now on [docs.microsoft.com](#)
Announcing the Fastest ASP.NET Yet, ASP.NET Core 1.1 RTM
Visual Studio 2015 Update 3
Visual Studio Code

.NET Across Windows/Web Platforms



Source: <http://blogs.msdn.com/b/dotnet/archive/2014/12/04/introducing-net-core.aspx>

Evolution of ASP and ASP .NET

Active
Server
Pages
(Classic
ASP)

ASP.NET
(Web
Forms)

ASP.NET
MVC
1/2/3/4/5

ASP.NET
~~MVC 6~~
Core MVC

ASP.NET
Web Pages

Unified
MVC, Web
API and
Web
Pages

ASP.NET
Web API

Names & Version Numbers

C#
7.0

ASP.NET Core 1.x
(runs on .NET Core 1.x
or .NET Framework 4.5.2+)

.NET
Framework
4.6

.NET Core 1.x

ASP.NET
~~MVC 6~~
Core
MVC

SignalR

C# 7.0 in VS2017

The screenshot shows a Microsoft Edge browser window with the following details:

- Title Bar:** What's New in C# 7 | M X
- Address Bar:** docs.microsoft.com/en-us/dotnet/articles/csharp/csharp-7
- Content Area:**
 - In this article:** A sidebar listing various C# 7 features: out variables, Tuples, Pattern matching, Ref locals and returns, Local functions, More expression-bodied members, Throw expressions, Generalized async return types, and Numeric literal syntax improvements.
 - Left Sidebar:** A navigation tree:
 - What's new in C#
 - What's new in C# 7** (selected)
 - What's new in C# 6
 - C# Interactive
 - C# Concepts
 - Using the .NET Compiler Platform
 - Bottom Buttons:** Download PDF

<https://docs.microsoft.com/en-us/dotnet/articles/csharp/csharp-7>

ASP.NET Core Roadmap

The screenshot shows a browser window displaying the [ASP.NET Core Schedule and Roadmap](https://github.com/aspnet/Home/wiki/Roadmap) page on GitHub. The page title is "ASP.NET Core Schedule and Roadmap". A paragraph explains that the schedule and roadmap are subject to change. Below this, a table lists releases and their time frames. A red box highlights the table. A note at the bottom states that references to yearly quarters (Q1, Q2, Q3, Q4) are calendar-based. To the right, there's a sidebar with links to "Pages 17", "NuGet feeds", "Contributing" (with "How to contribute" and "Engineering guidelines" sub-links), and "Roadmap". There are also "Clone this wiki locally" buttons for GitHub and a "Clone in Desktop" button.

Schedule

Release	Time frame*
1.0.3	Dec 2016
1.0.4	Feb 2017
1.2	Q2 2017

* References to yearly quarters (Q1, Q2, Q3, Q4) are calendar-based

<https://github.com/aspnet/Home/wiki/Roadmap>

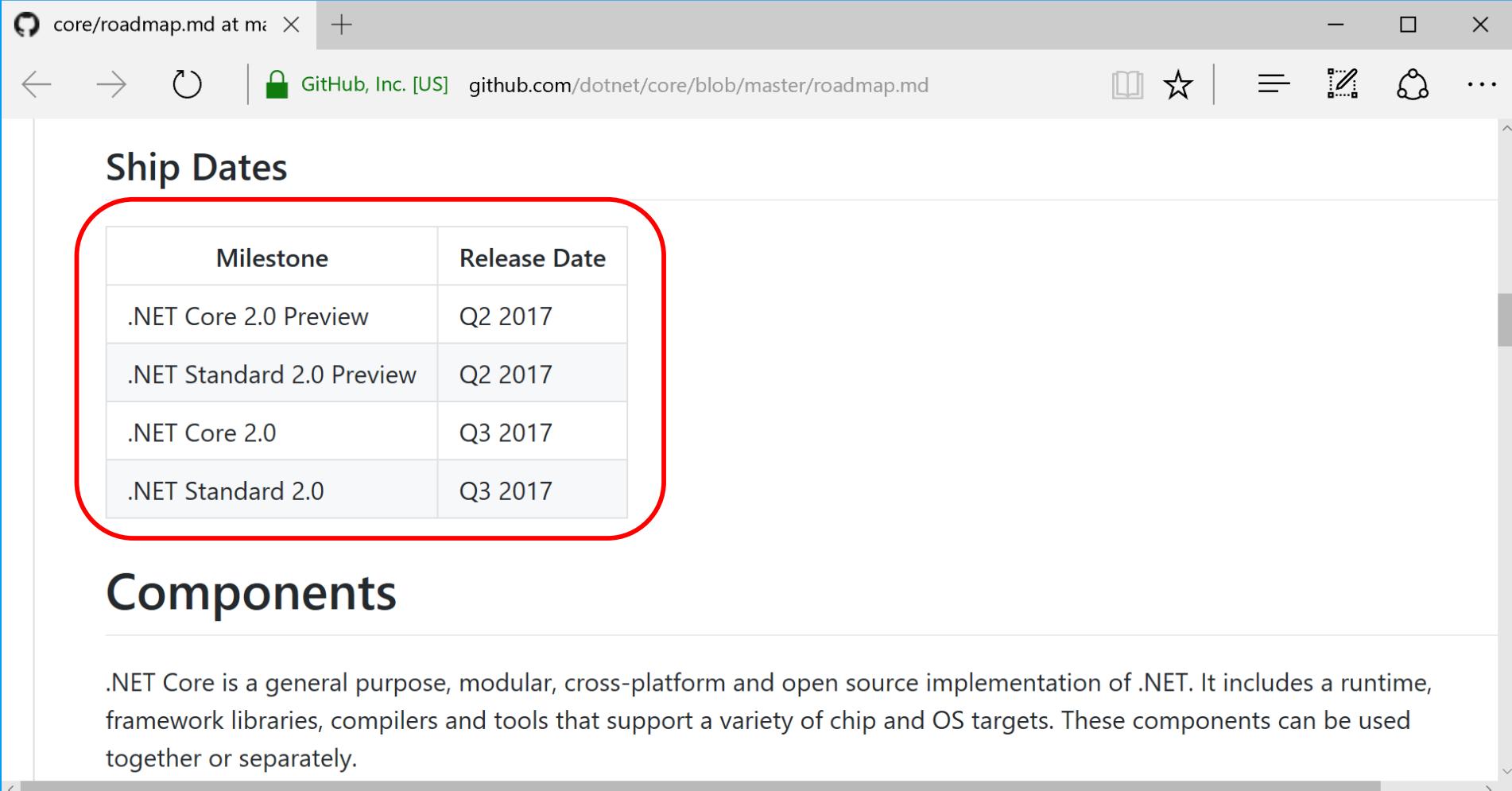
ASP.NET Core 1.1 Update

The screenshot shows a Microsoft Edge browser window with the following details:

- Title Bar:** "January 2017 Update fo X" and a "+" button.
- Address Bar:** "blogs.msdn.microsoft.com/dotnet/2017/01/30/january-2017-update-for-asp-net-core-1-1"
- Header:** "Server & Tools Blogs > Developer Tools Blogs > .NET Blog" and "Sign in".
- Navigation:** A purple navigation bar with dropdown menus for "Executive Bloggers", "Visual Studio", "Application Lifecycle Management", "Languages", ".NET", "Platform Development", and "Data Development".
- Section Headers:** ".NET Blog" and "Visual Studio".
- Post Title:** "January 2017 Update for ASP.NET Core 1.1" (highlighted with a red border).
- Post Details:** "January 30, 2017 by Rich Lander [MSFT] // 4 Comments".
- Share Buttons:** Facebook ("Share 30"), Twitter ("80"), LinkedIn ("36").
- Rating:** Five yellow stars.
- Social Links:** "Like 19K", "Follow @dotnet", "Follow @aspnet".
- Share This Post:** Buttons for Facebook, Twitter, LinkedIn, Email, and RSS.
- Page Footer:** A large blue footer bar with the Microsoft logo and "Microsoft .NET".

<https://blogs.msdn.microsoft.com/dotnet/2017/01/30/january-2017-update-for-asp-net-core-1-1/>

.NET Core Roadmap



core/roadmap.md at master · dotnet/core · GitHub

Ship Dates

Milestone	Release Date
.NET Core 2.0 Preview	Q2 2017
.NET Standard 2.0 Preview	Q2 2017
.NET Core 2.0	Q3 2017
.NET Standard 2.0	Q3 2017

Components

.NET Core is a general purpose, modular, cross-platform and open source implementation of .NET. It includes a runtime, framework libraries, compilers and tools that support a variety of chip and OS targets. These components can be used together or separately.

<https://github.com/dotnet/core/blob/master/roadmap.md>

.NET Core 2.0 (May 2017)

A screenshot of a Microsoft Edge browser window displaying the GitHub repository for .NET Core 2.0 Milestone 4. The URL in the address bar is <https://github.com/dotnet/corefx/milestone/4>. The page shows the repository details for `dotnet / corefx`, including metrics like 1,314 issues, 31 pull requests, and 0 projects. A prominent section highlights the '2.0.0' release, noting it is due by May 10, 2017, at 95% completion. A red box surrounds a note about the '.NET Core 2.0 Placeholder (next major release)' and a link to the roadmap. Below this, a progress bar indicates 256 open issues and 5,979 closed issues.

2.0.0

Due by May 10, 2017 95% complete

.NET Core 2.0 Placeholder (next major release)
<https://github.com/dotnet/core/blob/master/roadmap.md> Note: Previously known as 1.2.

256 Open 5,979 Closed

Update coreclr and solutions configurations Infrastructure Infrastructure

<https://github.com/dotnet/corefx/milestone/4>

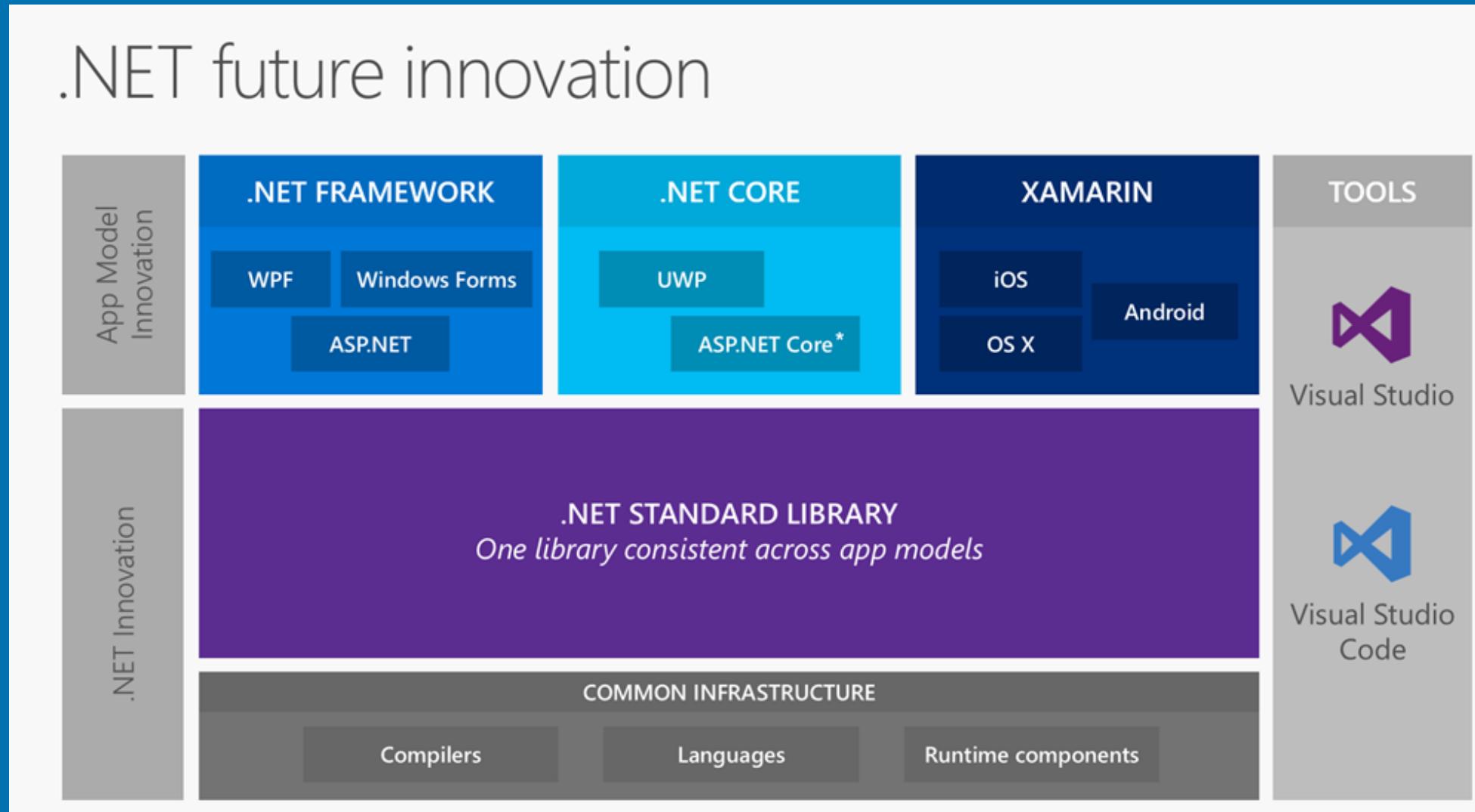
Announcing .NET Core Tools Updates in VS 2017 RC

The screenshot shows a Microsoft Edge browser window with the following details:

- Title Bar:** Announcing .NET Core
- Address Bar:** blogs.msdn.microsoft.com/dotnet/2017/02/07/announcing-net-core-tools-updates-in-vs-2017-rc
- Header:** Executive Bloggers, Visual Studio, Application Lifecycle Management, Languages, .NET, Platform Development, Data Development
- Section Headers:** .NET Blog, Visual Studio
- Text:** A first-hand look from the .NET engineering teams
- Post Title:** Announcing .NET Core Tools Updates in VS 2017 RC
- Post Date:** February 7, 2017 by Rich Lander [MSFT] // 48 Comments
- Share Buttons:** Facebook (253), Twitter (332), LinkedIn (98)
- Rating:** ★★★★☆
- Social Links:** Like 19K, Follow @dotnet, Follow @aspnet
- Share This Post:** Facebook, Twitter, LinkedIn, Email, RSS
- Search Bar:** Search MSDN with Bing
- Search Options:** Search this blog (radio button), Search all blogs (radio button)
- Text at Bottom:** The following improvements have been made in this release...

<https://blogs.msdn.microsoft.com/dotnet/2017/02/07/announcing-net-core-tools-updates-in-vs-2017-rc/>

.NET Standard Library + Tooling



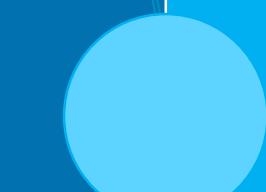
Source: <http://www.hanselman.com/blog/AnUpdateOnASPNETCore10RC2.aspx>

Agenda



Introduction

- > .NET (Framework & Core)
- > ASP.NET Core
- > Visual Studio

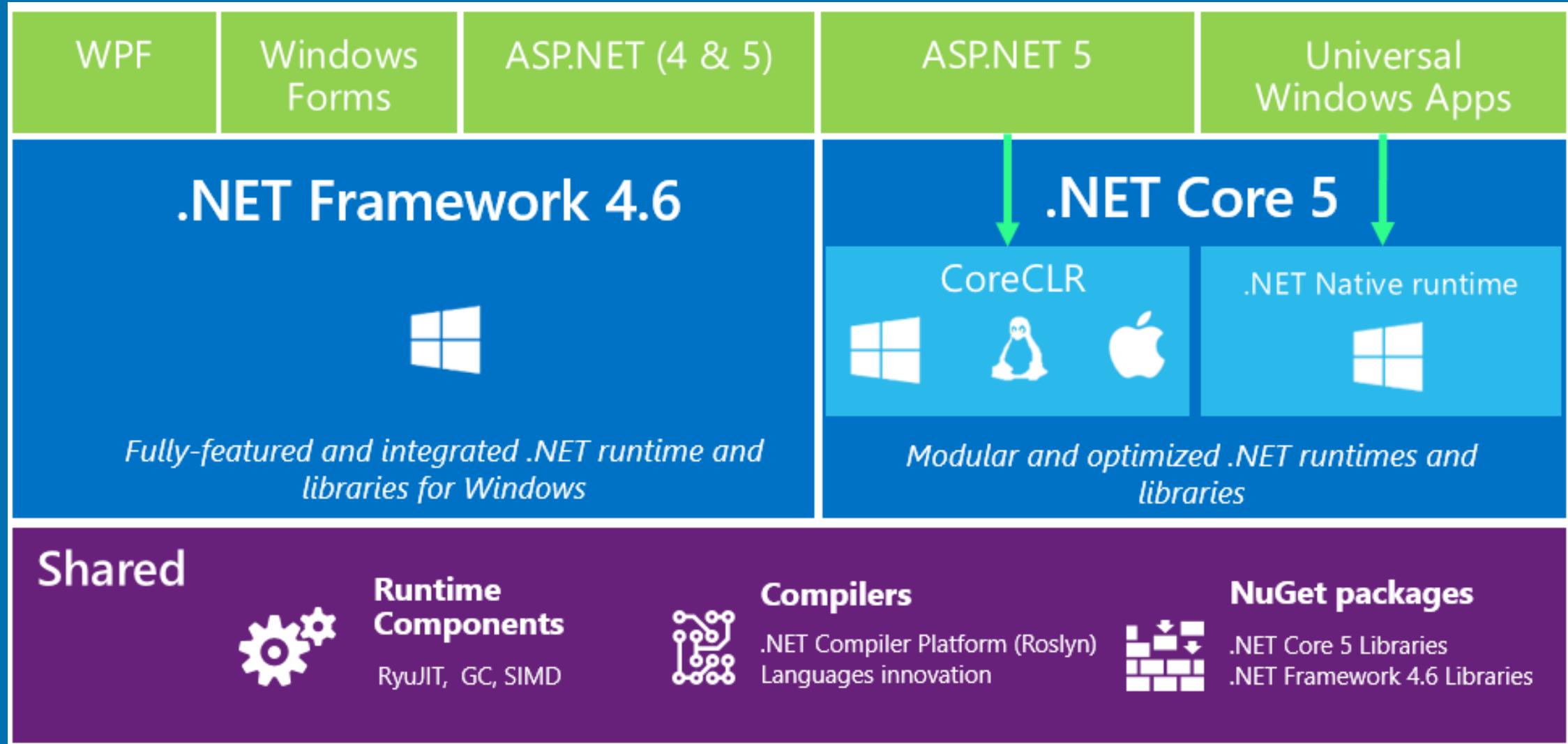


Q&A



.NET Framework & .NET Core

.NET in 2015: High-Level Overview



ASP.NET Core High-Level Overview

ASP.NET 4.6 and ASP.NET Core 1.0

ASP.NET 4.6

ASP.NET Core 1.0

.NET Framework 4.6



.NET Core 1.0

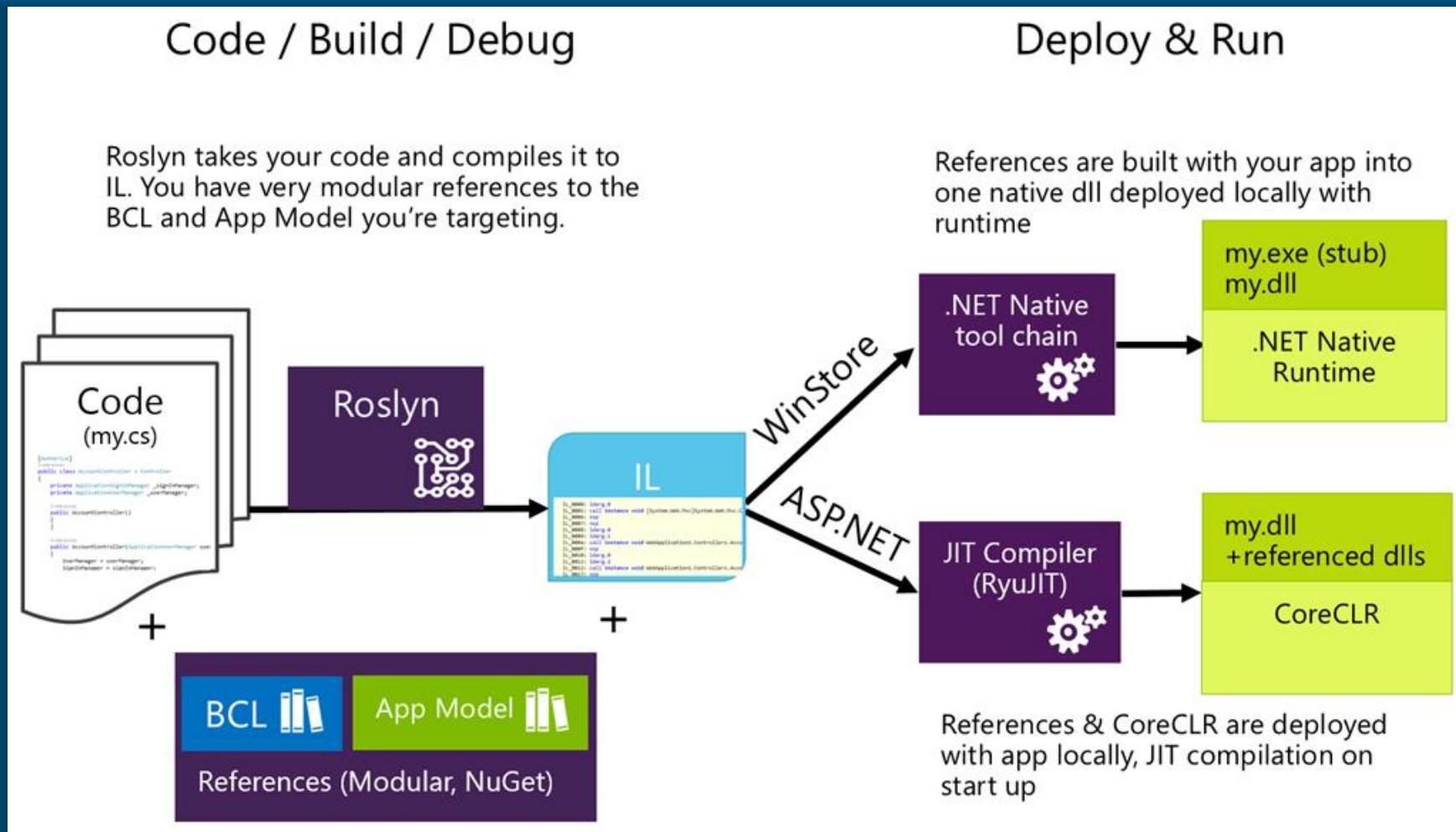


.NET framework libraries

.NET core libraries

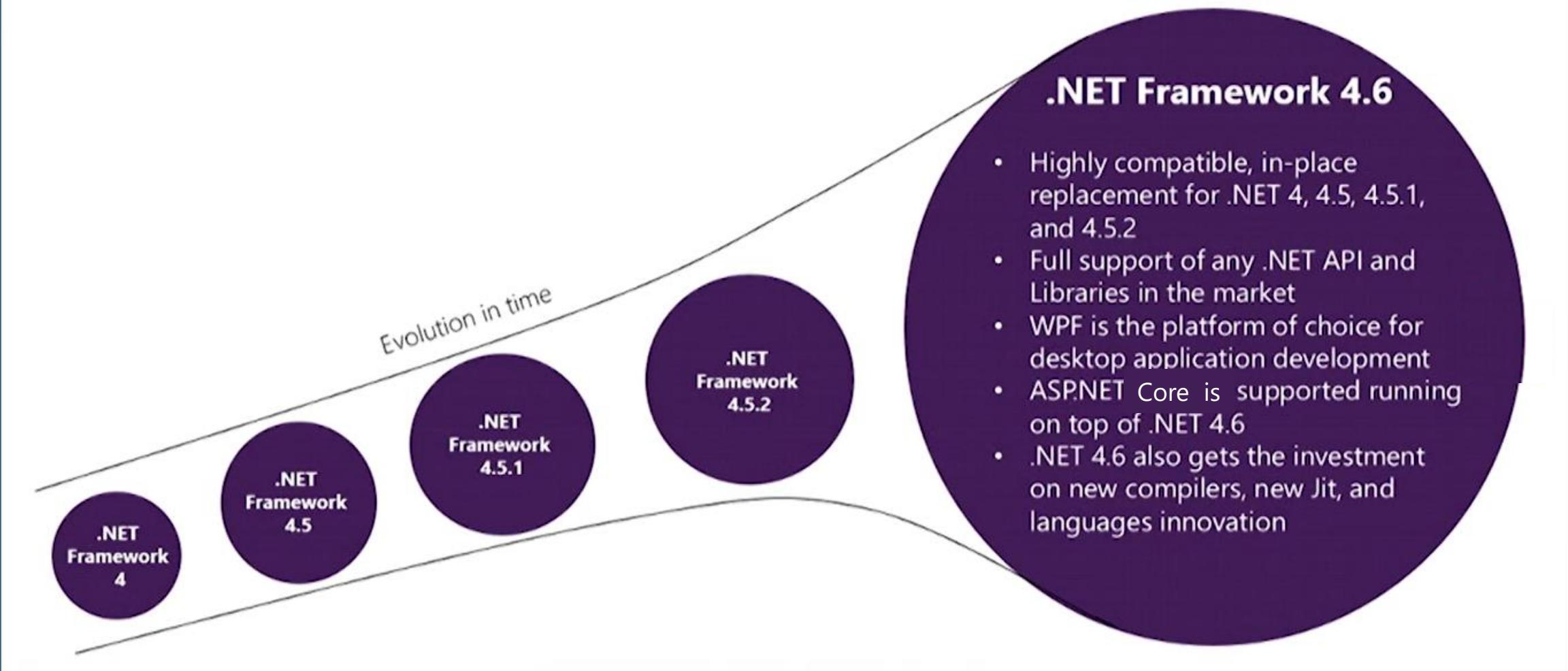
Compilers and runtime components
(.NET Compiler Platform: Roslyn, C#, VB, F# Languages, RyuJIT, SIMD)

Compilation Process



What About .NET Framework 4.6?

With over ~1B installations, we will continue to evolve .NET 4.x in a highly compatible manner



ASP .NET Core

ASP.NET Core 1.x Features

Cloud-ready

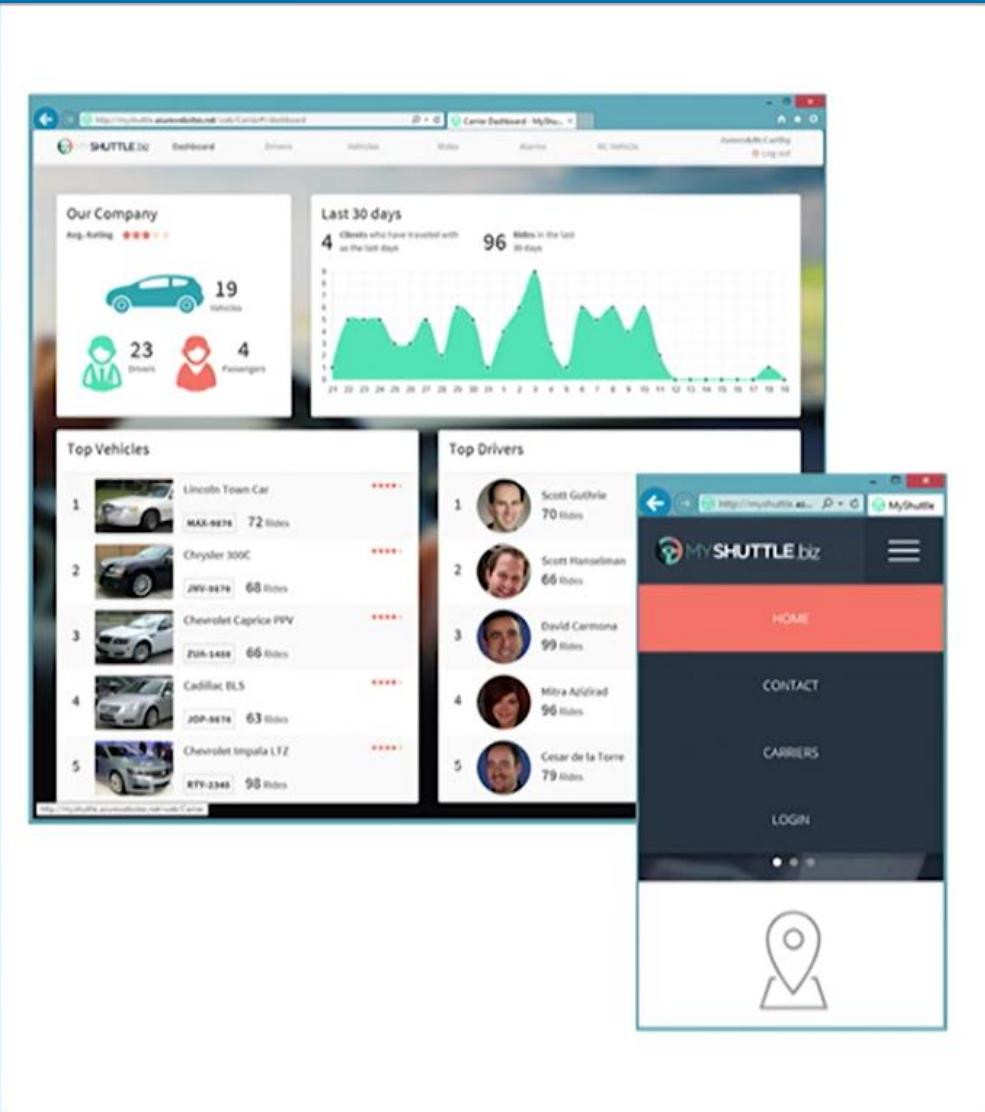
- Leaner, faster, simpler
- Designed from top to bottom to be ready for the cloud and cross-platform deployments

Modular and open

- More flexible with open source and modular implementation

Improved tooling and frameworks

- Deliver value faster with improved tooling and frameworks

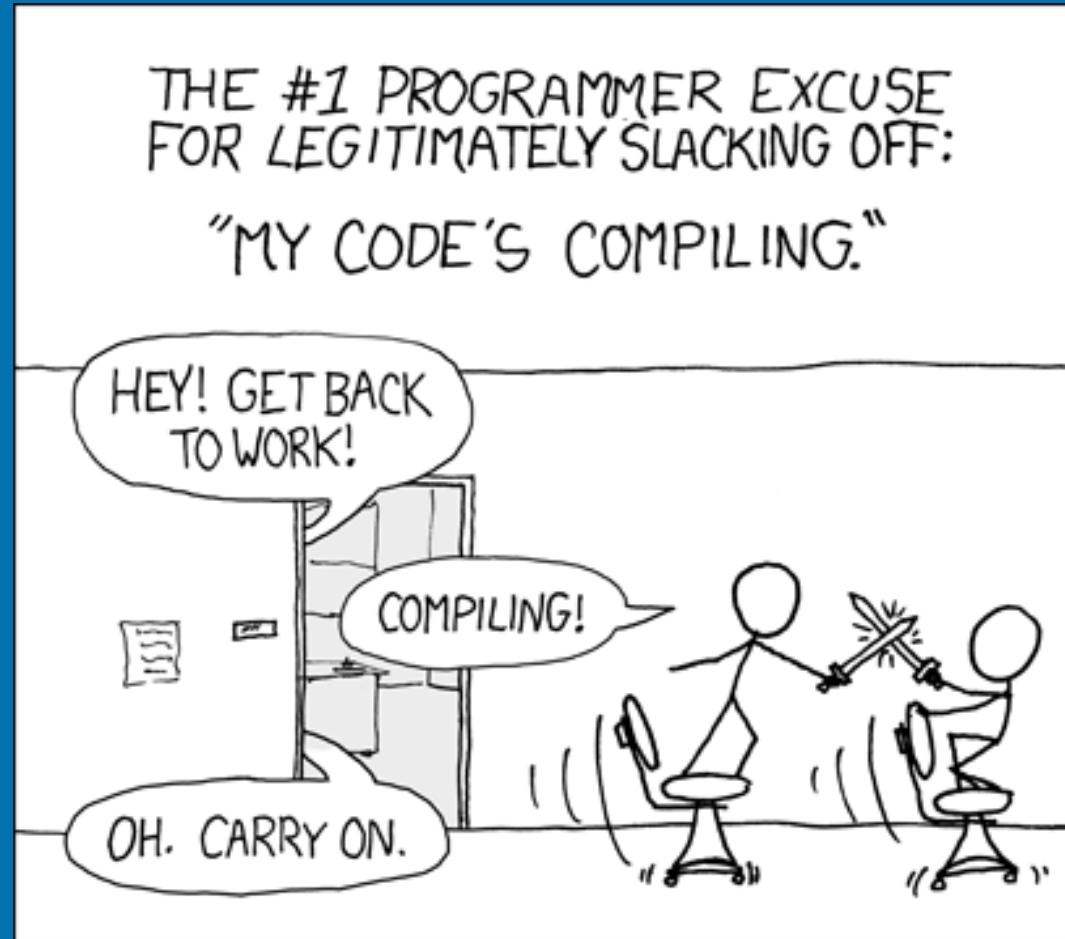


ASP.NET Core Summary

(MVC + Web API + Web Pages), SignalR 3

Feature	Running on .NET 4.6	Running on .NET Core
Cloud Ready	*	*
Modular Design	*	*
Dependency Injection	*	*
Consistent Tracing / Debugging	*	*
Faster Development (No Build Step)	*	*
Open Source	*	*
Full Side by Side (framework deployed inside application)		*
Fast startup, Low memory / High throughput (best of class)		*

Relevant XKCD Comic



Source: <https://xkcd.com/303/>

How about Entity Framework?

Data access / ORM investments in .NET

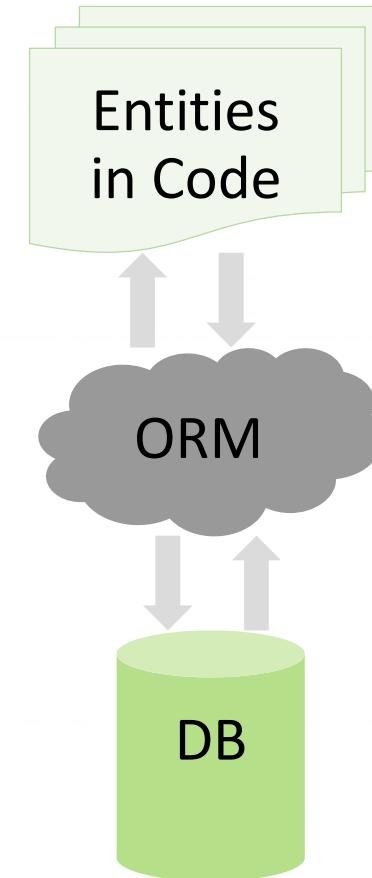
Entity Framework 6.x

- Evolution of current EF
- Runs on .NET Framework 4.6

Entity Framework Core

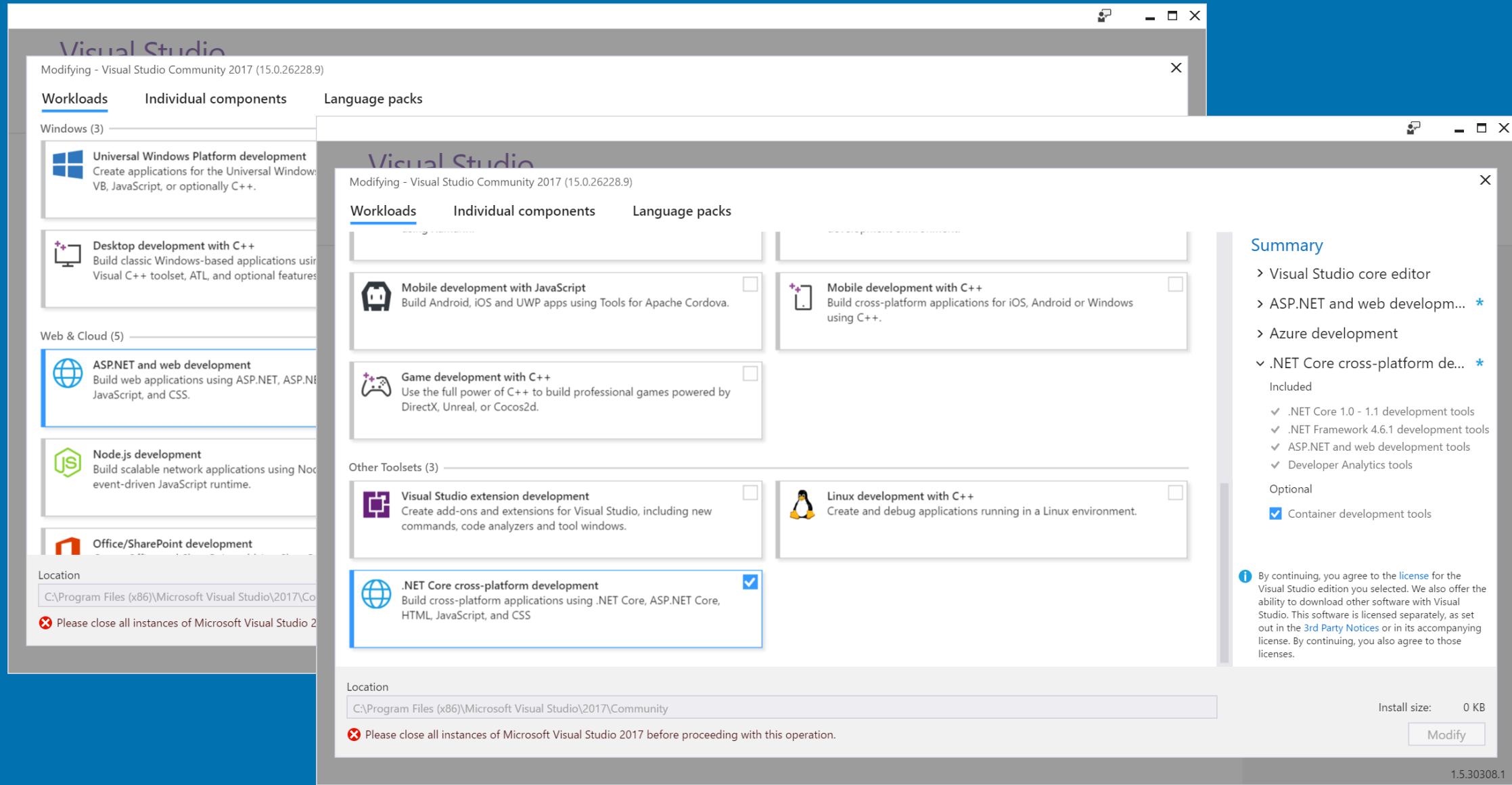
- LINQ and EF skills re-use, easy to move most existing apps
- Cloud optimized (small memory footprint and high throughput)
- Device optimized (lightweight, works with SQLite)
- Supports relational and non-relational stores
- Cross-platform ready (based on .NET Core)
- Design for disconnected/web scenarios
- Runs on .NET Core and .NET Framework 4.6

Both versions are OPEN SOURCE!

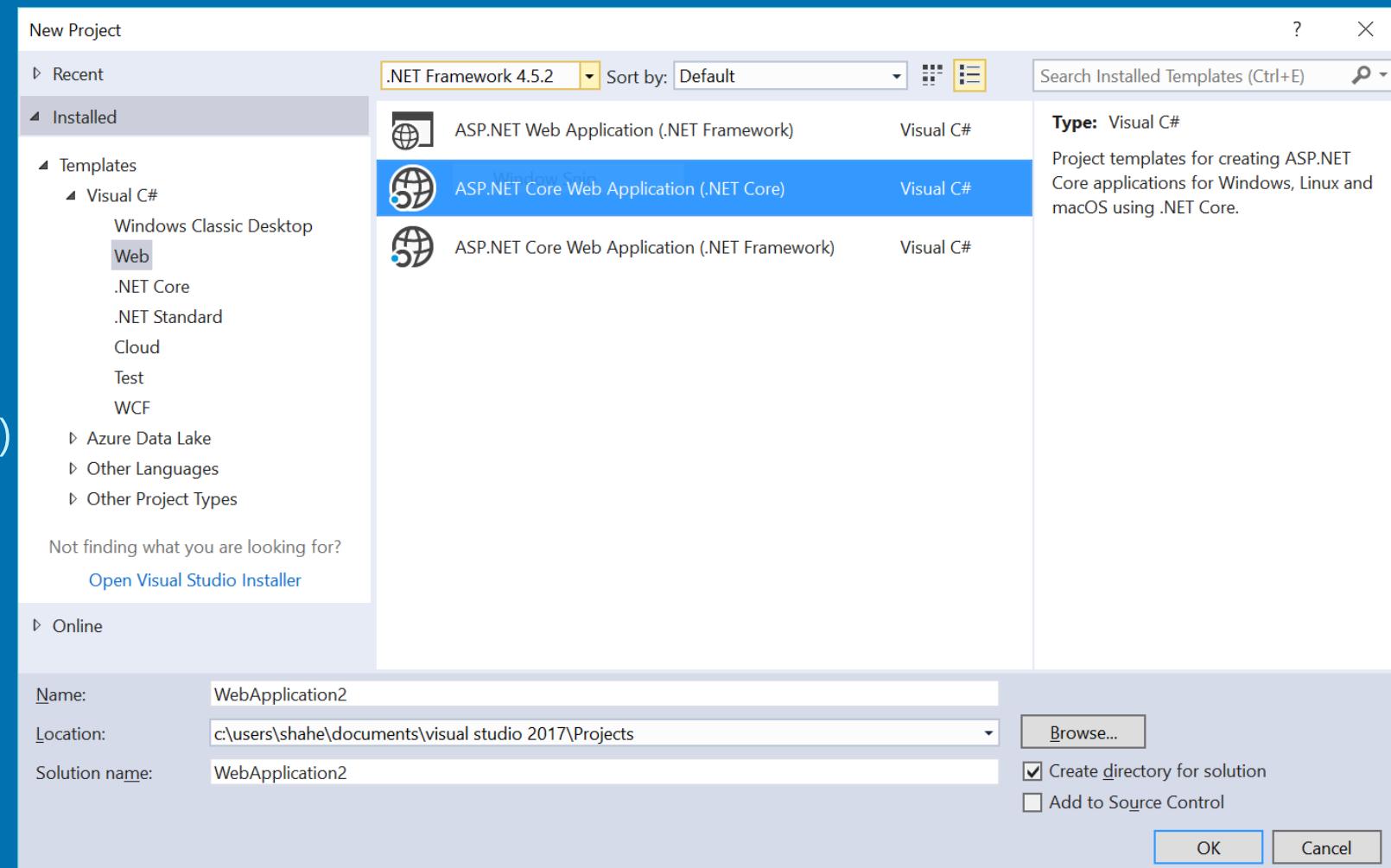


Visual Studio 2017

New Installer!

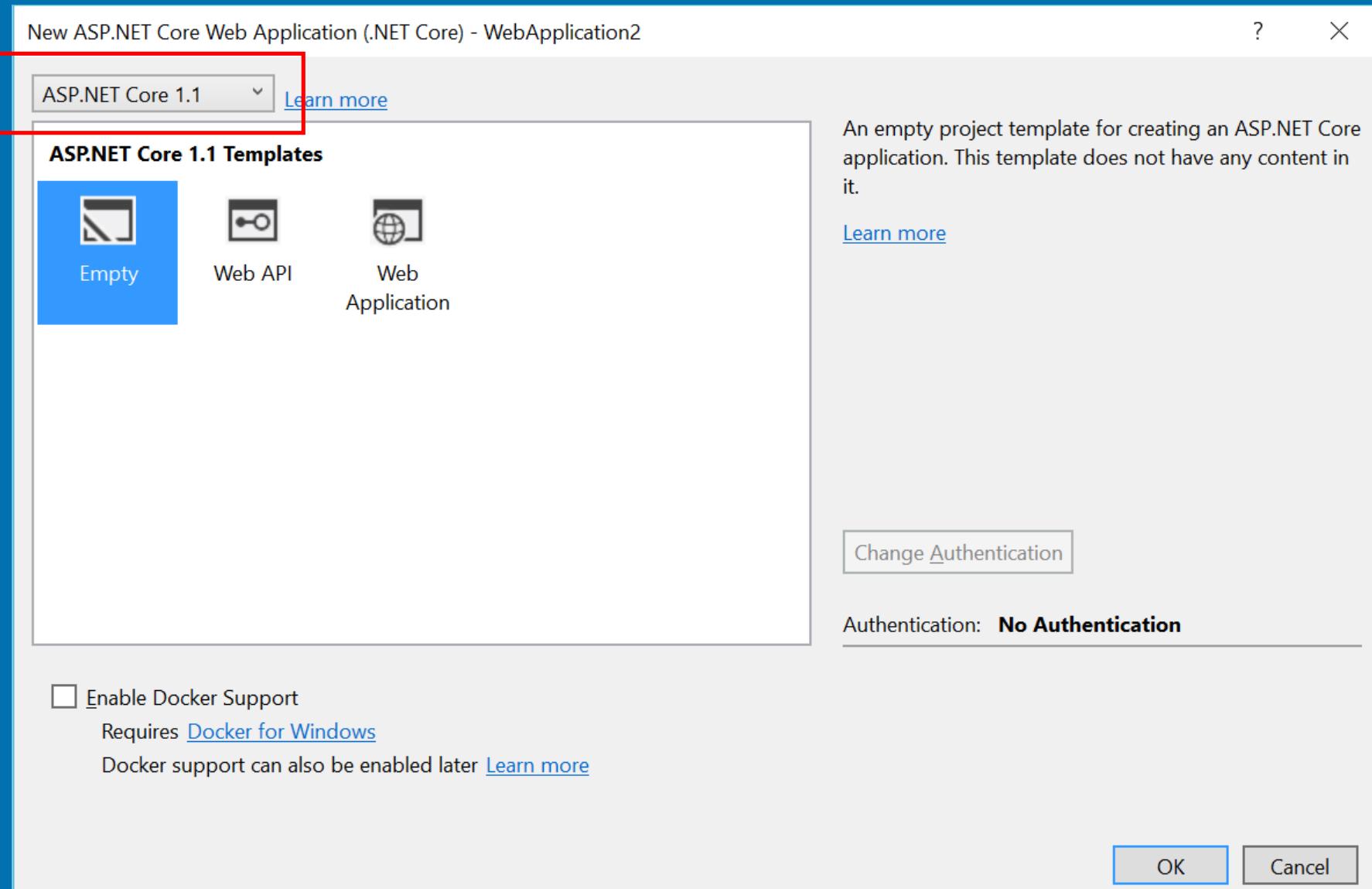


File → New Project → Web



- Web App (4.x)
- Core Web App (.NET Core)
- Core Web App (.NET framework)

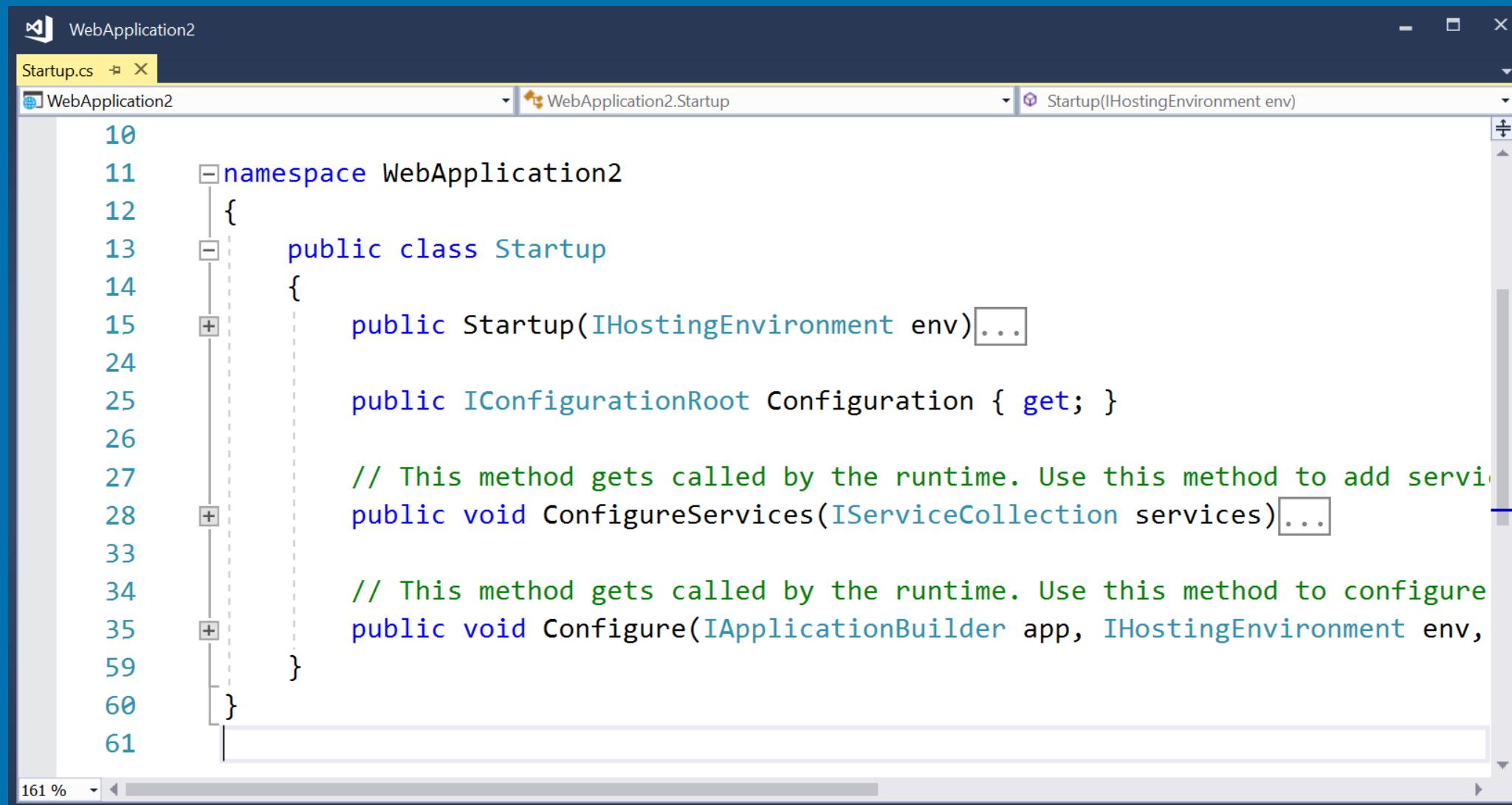
Select a Template



1.0 and 1.1 Templates

- Empty
- Web API
- Web Application

Startup.cs Configuration



The screenshot shows the Visual Studio IDE with the file `Startup.cs` open. The title bar indicates the project is `WebApplication2`. The code editor displays the following C# code:

```
10
11  namespace WebApplication2
12  {
13      public class Startup
14      {
15          public Startup(IHostingEnvironment env)...
16
17          public IConfigurationRoot Configuration { get; }
18
19          // This method gets called by the runtime. Use this method to add services to the container.
20          public void ConfigureServices(IServiceCollection services)...
21
22          // This method gets called by the runtime. Use this method to configure the HTTP request pipeline.
23          public void Configure(IApplicationBuilder app, IHostingEnvironment env,
24                               ILoggerFactory loggerFactory)
25          {
26              app.UseStaticFiles();
27
28              app.UseMvc();
29
30              loggerFactory.AddConsole(Configuration.GetSection("Logging"));
31              loggerFactory.AddDebug();
32          }
33      }
34  }
```

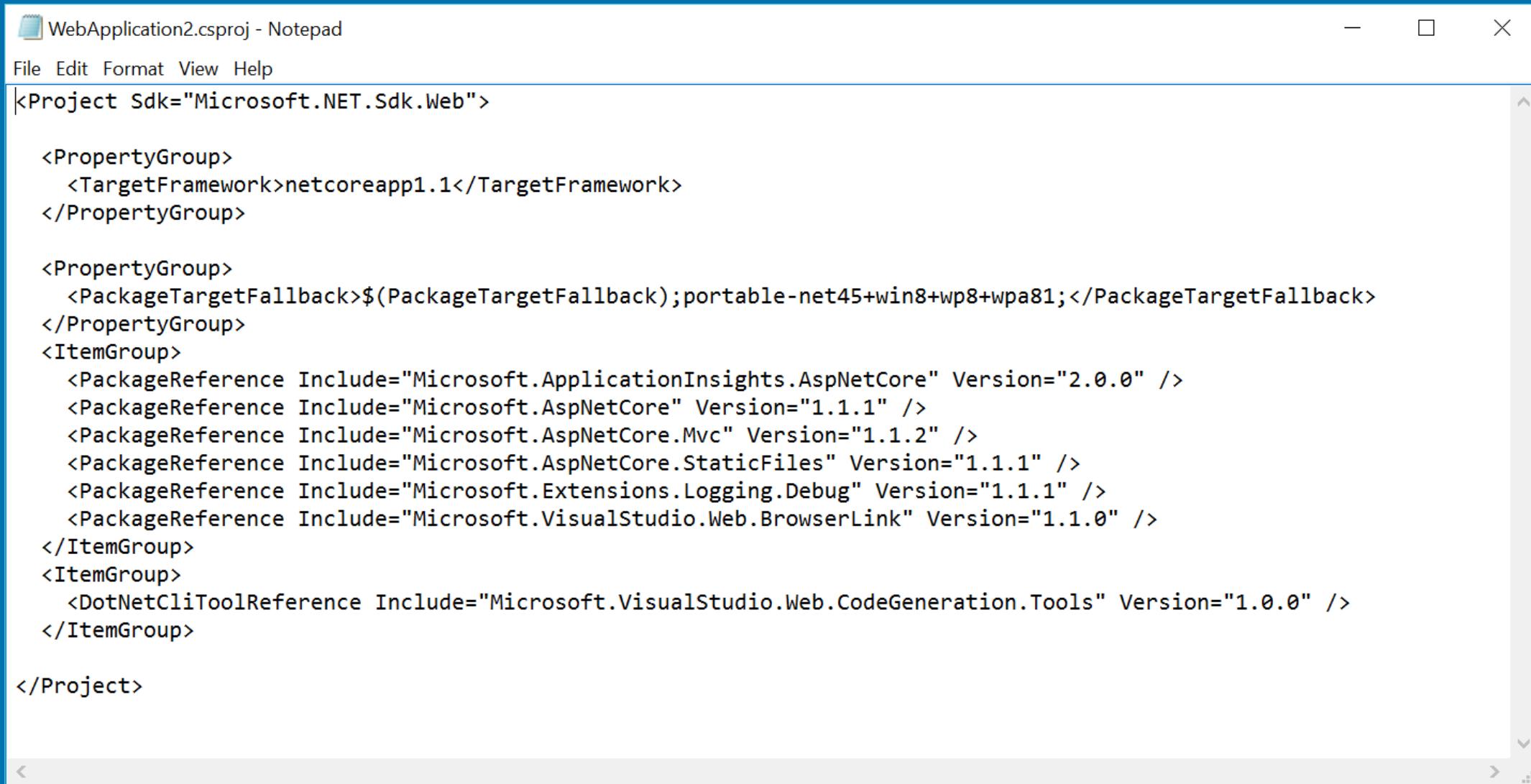
The code implements the `Startup` interface, which provides configuration and service setup logic for the application. The `ConfigureServices` method adds services to the container, and the `Configure` method configures the HTTP request pipeline.

project.json

A screenshot of a code editor window titled "project.json". The tab bar also includes "Startup.cs". Below the tabs, it says "Schema: http://json.schemastore.org/project". The main content area displays the JSON structure of the project file, with line numbers on the left and collapsible sections indicated by plus signs (+) next to some keys.

```
1  {-{
2      "userSecretsId": "aspnet-rc2web-fcef7c07-b6f5-4f2b-a926-e04d3a04731b",
3
4      "+ dependencies": [...],
43
44      "+ tools": [...],
72
73      "+ frameworks": [...],
82
83      "+ buildOptions": [...],
87
88      "+ runtimeOptions": [...],
91
92      "+ publishOptions": [...],
100
101     "+ scripts": [...]
105 }}
```

.csproj project file



The image shows a screenshot of a Windows Notepad window titled "WebApplication2.csproj - Notepad". The window contains the XML code for a .csproj project file. The code defines a project using the Microsoft.NET.Sdk.Web SDK, specifies a target framework of netcoreapp1.1, and includes various package references and tool references for an ASP.NET Core application.

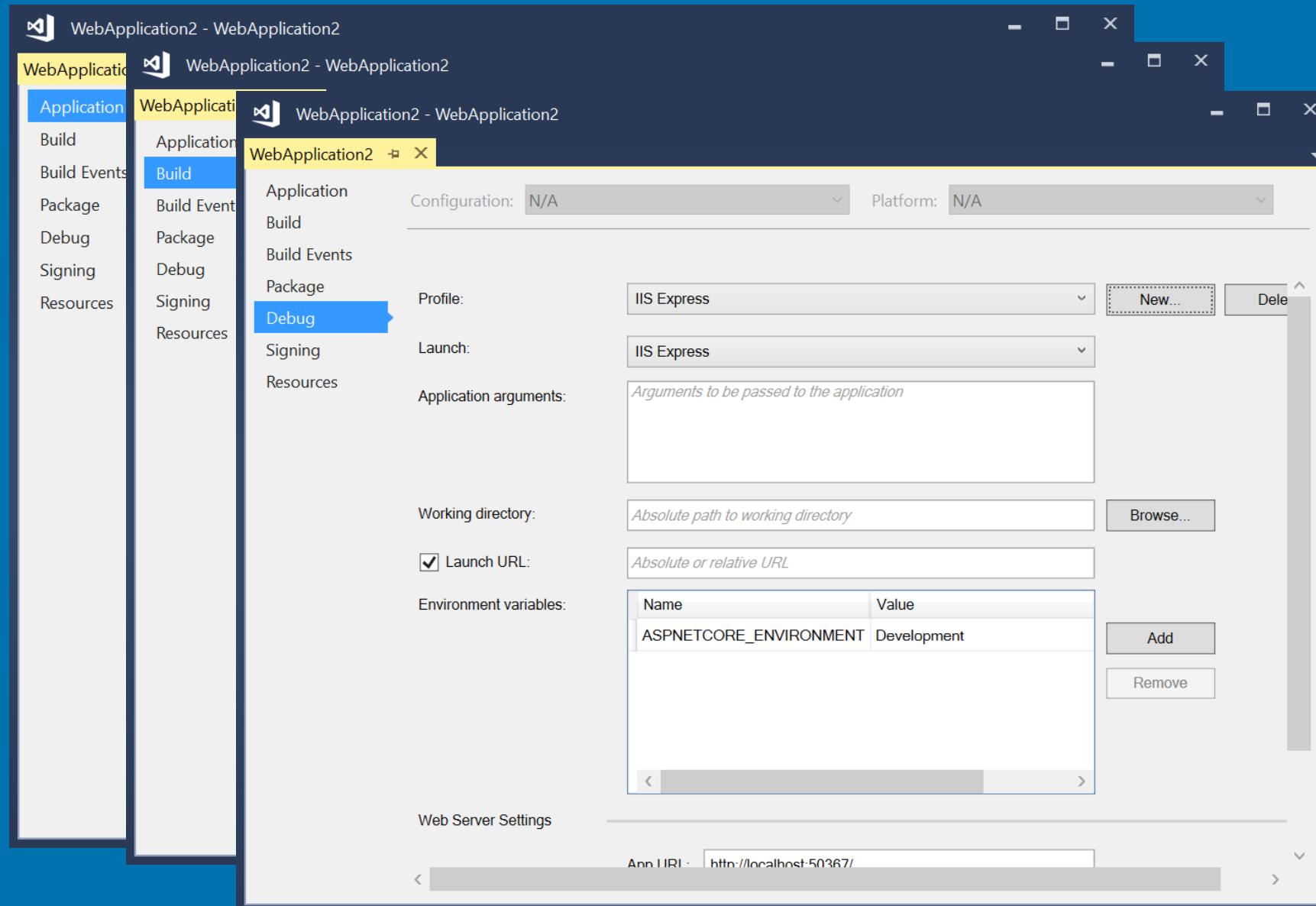
```
<Project Sdk="Microsoft.NET.Sdk.Web">

  <PropertyGroup>
    <TargetFramework>netcoreapp1.1</TargetFramework>
  </PropertyGroup>

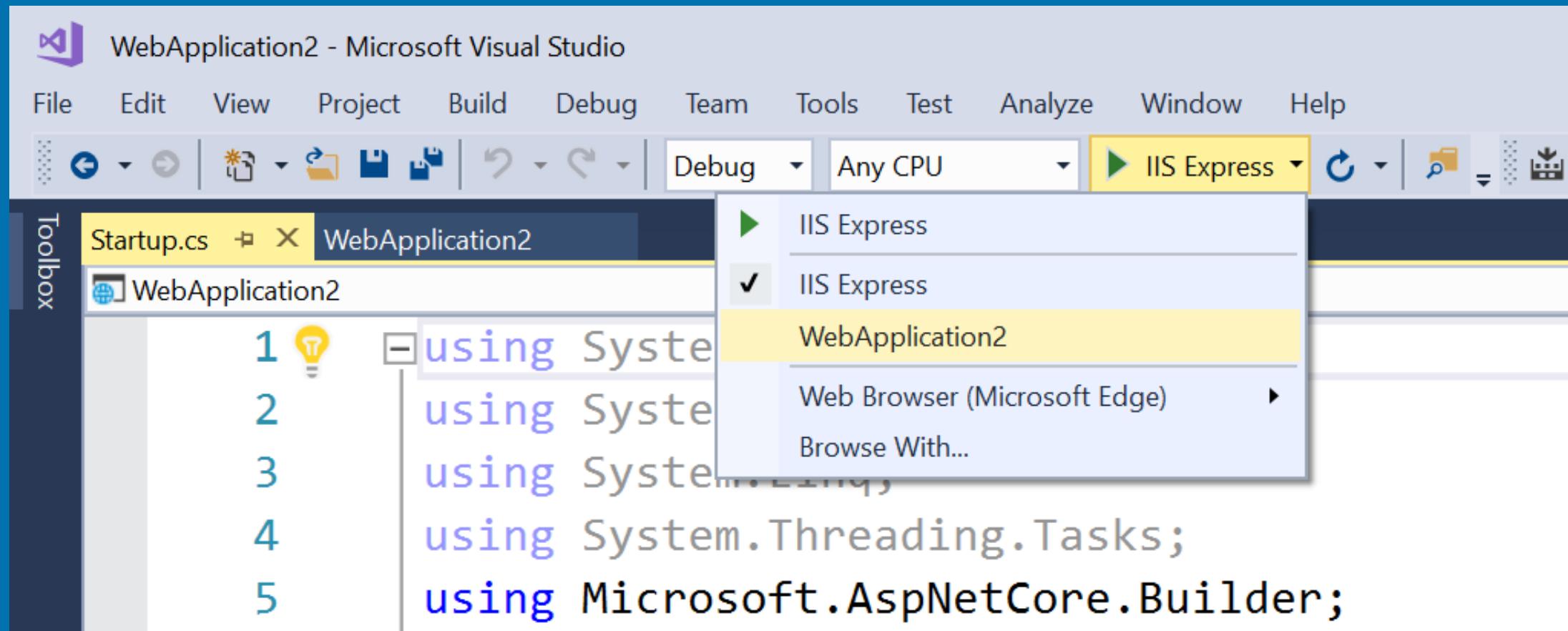
  <PropertyGroup>
    <PackageTargetFallback>$(PackageTargetFallback);portable-net45+win8+wp8+wpa81;</PackageTargetFallback>
  </PropertyGroup>
  <ItemGroup>
    <PackageReference Include="Microsoft.ApplicationInsights.AspNetCore" Version="2.0.0" />
    <PackageReference Include="Microsoft.AspNetCore" Version="1.1.1" />
    <PackageReference Include="Microsoft.AspNetCore.Mvc" Version="1.1.2" />
    <PackageReference Include="Microsoft.AspNetCore.StaticFiles" Version="1.1.1" />
    <PackageReference Include="Microsoft.Extensions.Logging.Debug" Version="1.1.1" />
    <PackageReference Include="Microsoft.VisualStudio.Web.BrowserLink" Version="1.1.0" />
  </ItemGroup>
  <ItemGroup>
    <DotNetCliToolReference Include="Microsoft.VisualStudio.Web.CodeGeneration.Tools" Version="1.0.0" />
  </ItemGroup>

</Project>
```

Right-click → (Project) Properties



Choose Profile While Debugging



New Razor Pages!

The screenshot shows a Microsoft Edge browser window displaying a blog post. The title of the post is "What are Razor Pages trying to do & What are not?". The URL in the address bar is "hishambinateya.com/welcome-razor-pages". The page content discusses the goals and non-goals of Razor Pages.

The razor pages are trying to:

- Make dynamic html and forms easier in ASP.NET Core.
- Reduce the number of folders and files in MVC structure that requires for page-focused scenarios.
- Simplify the code for page-focused pattern.
- Enable to return non html responses when necessary.
- Use MVC primitives as much as possible.
- Simplify the migration to traditional MVC structure.

The razor pages are not trying to:

- Create scripted page framework to compete with PHP .. etc.
- Hide C# with Domain Specification Language (DSL) in razor or otherwise.
- Create new primitives that only applicable for this model.

http://www.hishambinateya.com/welcome-razor-pages

New: Live Unit Testing

Live Unit Testing experience

Once enabled, Live Unit Testing helps you quickly see whether the code you're writing is covered and if the tests that cover it are passing, without leaving the editor. Unit test results and coverage visualizations appear on a line-by-line basis in the code editor as shown in sample image below:

- ✖ If a line of executable code is covered by at least one failing test, Live Unit Testing will decorate it with a red "x".
- ✓ If a line of executable code is covered by only passing tests, Live Unit Testing will decorate it with a green "√".
- If a line of executable code is not covered by any test, Live Unit Testing will decorate it with a blue dash.

Trainer.cs

```
35     int count = 0;
36     foreach (var work in _workouts)
37     {
38         count += work.Miles;
39     }
40     return count;
41 }
42
43 private List<Workout> _workouts;
44
45 public Trainer(int goal)
46 {
47     _workouts = new List<Workout>();
48     Goal = goal;
49 }
50
51 public void RegisterWorkout(int miles, TimeSpan duration)
52 {
53     _workouts.Add(new Workout(miles, duration));
54 }
55
56 public bool HasMetGoal()
57 {
58 }
```

DEMO

The screenshot shows a Microsoft Edge browser window with the title bar "Home Page - WebAppl" and address bar "localhost:50367". The page content is the ASP.NET Core documentation home page. At the top, there's a navigation bar with "WebApplication2" and links for "Home", "About", and "Contact". Below the navigation bar is a large blue banner with the text "ASP.NET Core" on the left, followed by a vertical separator line, and "Windows", "Linux", and "OSX" on the right. In the center of the banner is the text "Learn how to build ASP.NET apps that can run anywhere." with a "Learn More" button. Below the banner is a navigation menu with four sections: "Application uses", "How to", "Overview", and "Run & Deploy", each with a list of bullet points.

Application uses

- Sample pages using ASP.NET Core MVC
- Bower for managing client-side libraries
- Theming using Bootstrap

How to

- Add a Controller and View
- Manage User Secrets using Secret Manager.
- Use logging to log a message.
- Add packages using NuGet.
- Add client packages using Bower.
- Target development, staging or production environment.

Overview

- Conceptual overview of what is ASP.NET Core
- Fundamentals of ASP.NET Core such as Startup and middleware.
- Working with Data
- Security
- Client side development
- Develop on different platforms
- Read more on the documentation site

Run & Deploy

- Run your app
- Run tools such as EF migrations and more
- Publish to Microsoft Azure Web Apps

Migrating from MVC to MVC Core

The screenshot shows a Microsoft Edge browser window with the title bar "Migrating From ASP.NET X". The address bar displays the URL "docs.microsoft.com/en-us/aspnet/core/migration/mvc". The main content area is titled "Migrating From ASP.NET MVC to ASP.NET Core MVC". Below the title, it says "3/7/2017 • 7 min to read • Contributors" followed by five small profile icons and a "all" link. A sidebar on the left contains a "Filter" input field and a list of topics: Testing, Working with Data, Client-Side Development, Mobile, Publishing and Deployment, Guidance for Hosting Providers, Security, Performance, and Migration. The "Migration" topic is currently selected, indicated by a blue underline. At the bottom of the sidebar are links for "Download PDF" and "Feedback". The main content area features a section titled "In this article" with links to various migration steps: Create the starter ASP.NET MVC project, Create the ASP.NET Core project, Configure the site to use MVC, Add a controller and view, Controllers and views, Static content, Migrate the layout file, Configure Bundling & Minification, Solving HTTP 500 errors, and Additional Resources.

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

Visual Studio Code

The image shows a screenshot of the Visual Studio Code website at code.visualstudio.com and a screenshot of the Visual Studio Code application window.

Visual Studio Code Website:

- Header:** Visual Studio Code - Co X +
- Address Bar:** code.visualstudio.com
- Navigation:** Visual Studio Code, Docs, Updates, Blog, Extensions, FAQ
- Search:** Search Docs
- Download:** Download

Content:

- Hero Section:** "Code editing. Redefined."
- Text:** Free. Open source. Runs everywhere.
- Download Buttons:** Download for Windows (Stable Build), Other platforms and Insiders Edition
- Legal Note:** By using VS Code, you agree to its license and privacy statement.

Visual Studio Code Application:

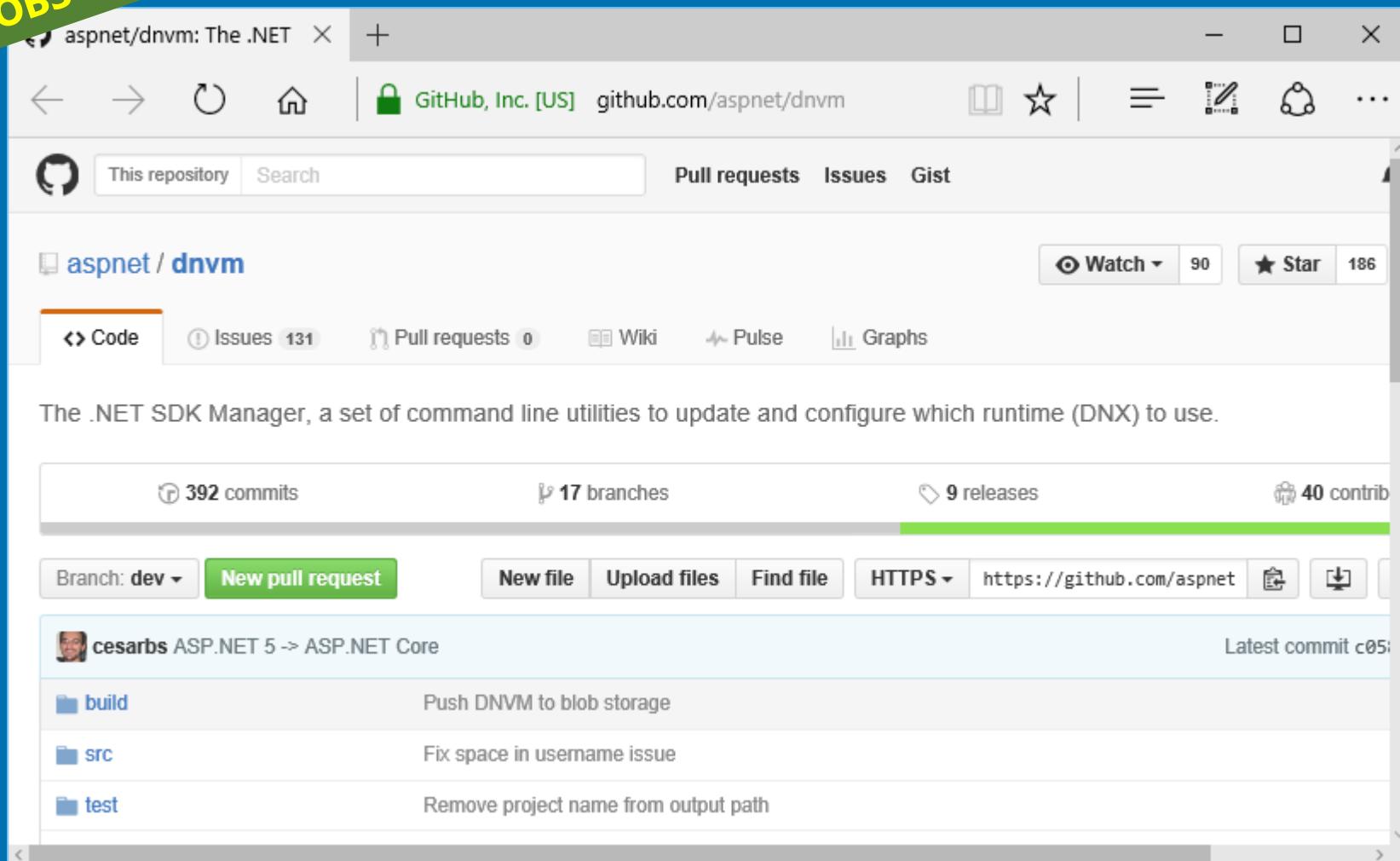
- Title Bar:** www.ts - node-express-ts - Visual Studio Code
- File Menu:** File, Edit, View, Goto, Help
- Sidebar:** EXTENSIONS, @popular, C#, Python, Debugger for Chrome, C/C++, Go, ESLint
- Code Editor:** app.ts, www.ts, package.json, README.md
- Code:**

```
1 import app from './app';
2 import debugModule = require('debug');
3 import http = require('http');
4
5 const debug = debugModule('node-express-typescript:server');
6
7 // Get port from environment and store in Express.
8 const port = normalizePort(process.env.PORT || '3000');
9 app.set('port', port);
10
11 // create
12 const server = app.listen(port);
13 server.on('listening', () => {
14   console.log(`Server listening on port ${port}`);
15 });
16
17 /**
18 * Normalize port: get a number, string, or boolean and return it
19 * as a number. Handle non-number types by falling back to
20 * a boolean: true means 'use process.env.PORT', false means 'use
21 * port'.
22 */
23 function normalizePort(val: any): number|string|boolean {
24   let port = parseInt(val, 10);
25
26   if (isNaN(port)) {
27     if (val === '') {
28       return undefined;
29     }
30     else if (val === 'true') {
31       return true;
32     }
33     else if (val === 'false') {
34       return false;
35     }
36     else {
37       throw new Error(`Could not parse port: ${val}`);
38     }
39   }
40   else {
41     return port;
42   }
43 }
```
- Status Bar:** master, 11 131, 0 0, Ln 9, Col 21, Spaces: 2, UTF-8, LF, TypeScript, Smiley icon

Download: <https://code.visualstudio.com>

.NET Version Manager (DNVM)

(OBSOLETE)



- .NET SDK Version Manager
- Formerly KVM
- Get list of DNXs (aka KRE)

GitHub: <https://github.com/aspnet/dnvm>

Tooling Changes and .NETStandard

The screenshot shows a web browser window with the URL hanselman.com/blog/AnUpdateOnASPNETCore10RC2.aspx. The page content discusses tooling changes, specifically mentioning the absence of dnvm replacement and the need for Visual Studio and OmniSharp to support the new CLI. A callout box highlights the first point. Another callout box contains a quote about the .NET CLI being the most destabilizing change in RC2. A third callout box contains the heading 'MOVING TO NETSTANDARD'.

- Tooling
 - There's no dnvm replacement
 - Visual Studio Tooling (UI) support needs to use the new CLI
 - **OmniSharp** needs to use the new CLI
 - What's the dnx-watch successor?

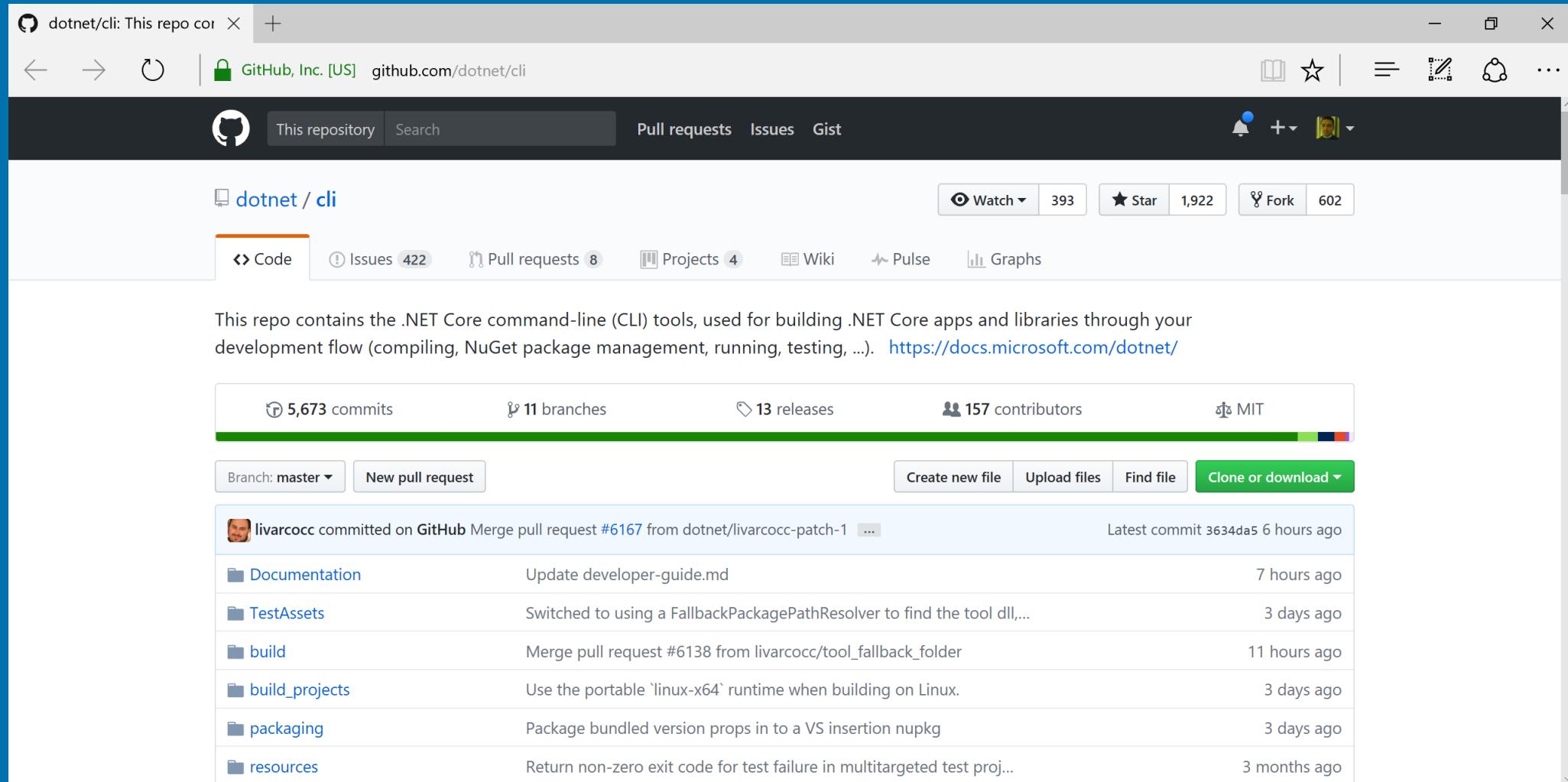
The list goes on and on. I'd suggest watching the [ASP.NET Community stand up](#) as we're pretty transparent about where we are in the process. We just got everyone internally using builds of Visual Studio that have CLI support this last week.

The new .NET CLI (again, replacing DNX) will be the most de-stabilizing change in RC2. This is a good intro to where things are headed <https://vimeo.com/153212604>. There's been tons of changes since then but it's still a good overview.

MOVING TO NETSTANDARD

This has been a long time coming and is a massive effort to get class library authors to move to the next phase of PCL. This is critical to get right so that everyone can have their favorite packages working on .NET Core, and as such, working everywhere.

dotnet/cli on GitHub



The screenshot shows the GitHub repository page for 'dotnet/cli'. The page title is 'dotnet/cli' and it's described as 'This repo contains the .NET Core command-line (CLI) tools, used for building .NET Core apps and libraries through your development flow (compiling, NuGet package management, running, testing, ...). <https://docs.microsoft.com/dotnet/>'.

Key statistics displayed on the page include:

- 5,673 commits
- 11 branches
- 13 releases
- 157 contributors
- MIT license

The 'Code' tab is selected. A recent commit by 'livarcocc' is highlighted: 'Merge pull request #6167 from dotnet/livarcocc-patch-1'. Other recent commits listed include:

- Documentation: Update developer-guide.md (7 hours ago)
- TestAssets: Switched to using a FallbackPackagePathResolver to find the tool dll... (3 days ago)
- build: Merge pull request #6138 from livarcocc/toolFallback_folder (11 hours ago)
- build_projects: Use the portable `linux-x64` runtime when building on Linux. (3 days ago)
- packaging: Package bundled version props in to a VS insertion nupkg (3 days ago)
- resources: Return non-zero exit code for test failure in multitargeted test proj... (3 months ago)

This repo contains the .NET Core command-line (CLI) tools, used for building .NET Core apps and libraries.

GitHub: <https://github.com/dotnet/cli>

References

- ASP.NET vNext: <http://asp.net/vnext>
- Intro: <https://weblogs.asp.net/scottgu/introducing-asp-net-5>
- Feb 2015 Updates: <http://blogs.msdn.com/b/webdev/archive/2015/02/23/aspnet-5-updates-for-feb-2015.aspx>
- dotnetConf March 2015: <https://channel9.msdn.com/Events/dotnetConf/2015>
- Build 2015: <https://channel9.msdn.com/Events/Build/2015>
- Understanding .NET 2015:
<http://blogs.msdn.com/b/bethmassi/archive/2015/02/25/understanding-net-2015.aspx>
- Grunt & Bower: <http://www.asp.net/vnext/overview/aspnet-vnext/grunt-and-bower-in-visual-studio-2015>
- Tutorial: <http://www.asp.net/vnext/overview/aspnet-vnext/vc>
- ASP.NET Community Standup Notes: <https://blogs.msdn.microsoft.com/webdev/>
- Update on ASP.NET RC2: <http://www.hanselman.com/blog/AnUpdateOnASPNETCore10RC2.aspx>
- Additional Tutorials: See Starter Web Project Template

Other Resources

- Roadmap: <https://github.com/aspnet/Home/wiki/Roadmap>
- ASP.NET Identity: <http://www.asp.net/identity/overview/getting-started/introduction-to-aspnet-identity>
- KRE, KVM, KPM: <http://gunnarpeipman.com/2014/10/asp-net-5-what-are-kre-kvm-kpm/>
- Grunt, JS Task Runner: <http://gruntjs.com/>
- Gulp, Workflow Automation: <http://gulpjs.com/>
- Bower, Package Manager: <http://bower.io/>
- npm, Node Package Manager: <https://www.npmjs.com/>

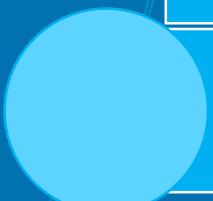
Agenda



Introduction



- > .NET (Framework & Core)
- > ASP.NET Core
- > Visual Studio



Q&A

Q & A

Agenda



Introduction



> .NET (Framework & Core)



> ASP.NET Core

> Visual Studio

Q&A

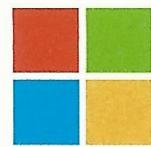
Shahed Chowdhuri

Sr. Technical Evangelist
US - DX Audience East

shchowd@microsoft.com

202-257-0444
<http://WakeUpAndCode.com>
@shahedC

Microsoft Corporation
12012 Sunset Hills Rd
Ste 100
Reston VA 20190



Microsoft

Email: shchowd@microsoft.com • Twitter: [@shahedC](https://twitter.com/@shahedC)