



Microsoft
Innovation
University

The logo consists of large, bold, white letters spelling "MIU". A white graduation cap is positioned above the letter "I". Inside the cap is a yellow lightbulb with a glowing filament, symbolizing knowledge or ideas.

Microsoft Innovation University

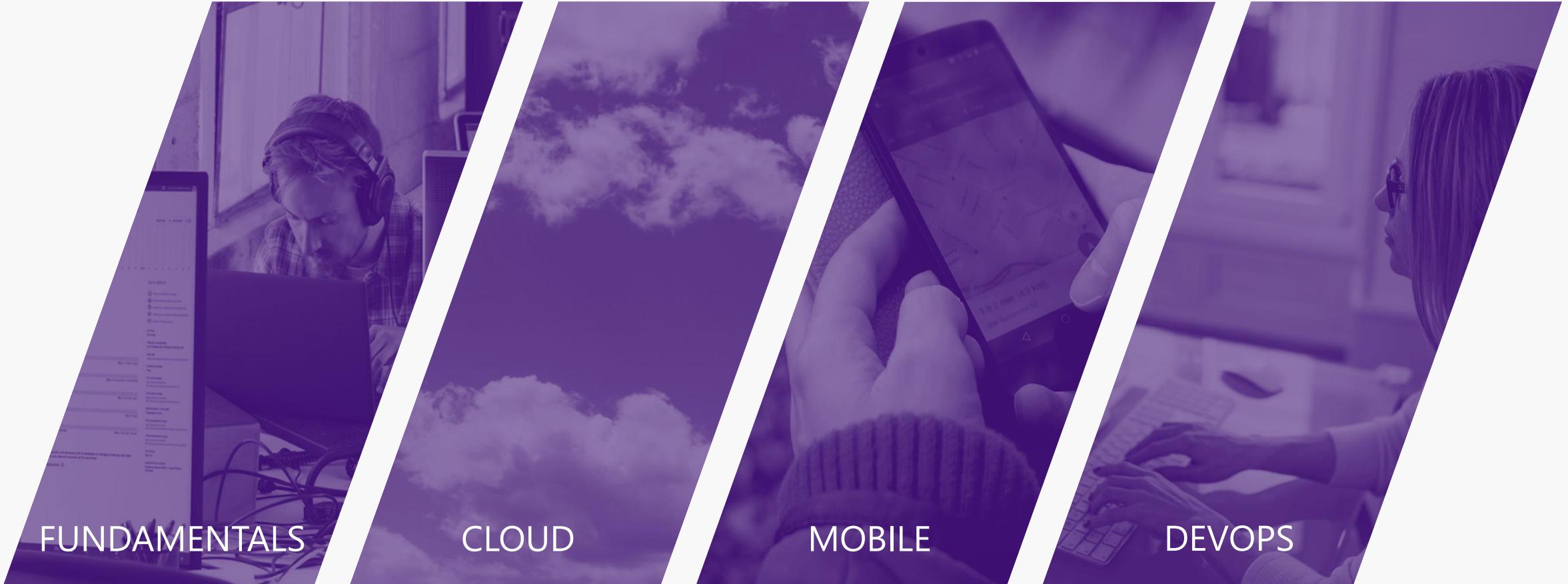


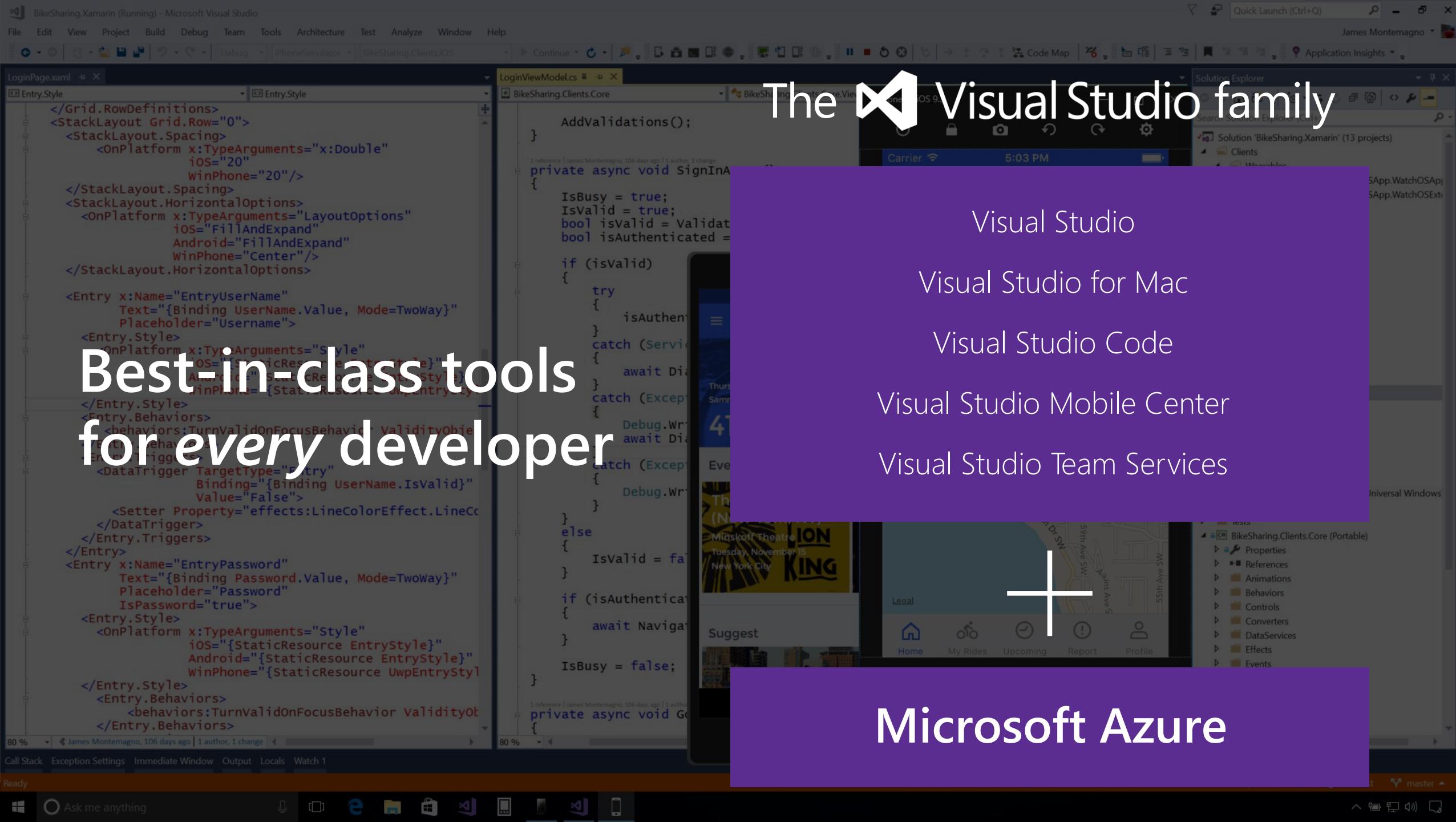
THE WORLD'S GREATEST SELECTION OF LUXURY

What's new in Visual Studio 2017 + Containers

João Almeida
joalmeid@microsoft.com
 @jalmeida

Visual Studio 2017





Best-in-class tools
for every developer

The  Visual Studio family

Visual Studio

Visual Studio for Mac

Visual Studio Code

Visual Studio Mobile Center

Visual Studio Team Services

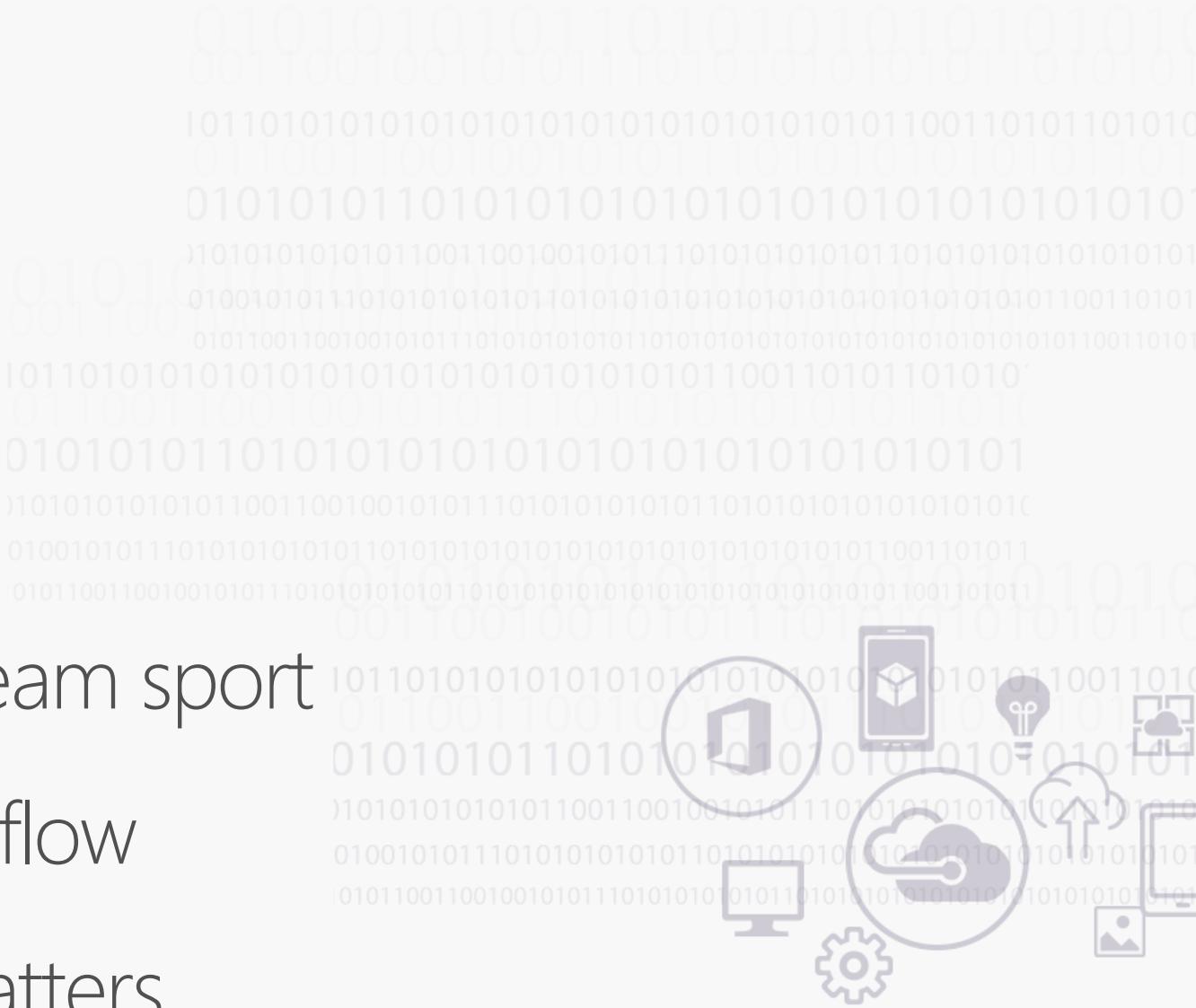
Microsoft Azure

Trends

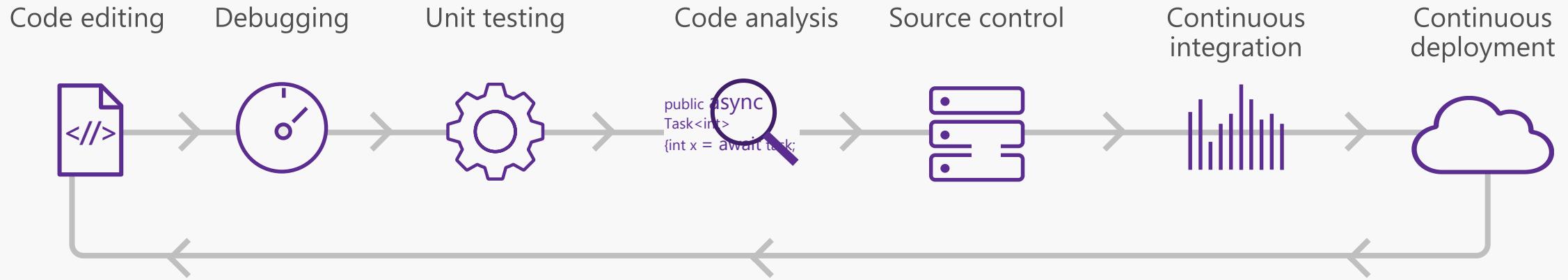
Software development is a team sport

Agile & DevOps change the flow

Individual productivity still matters

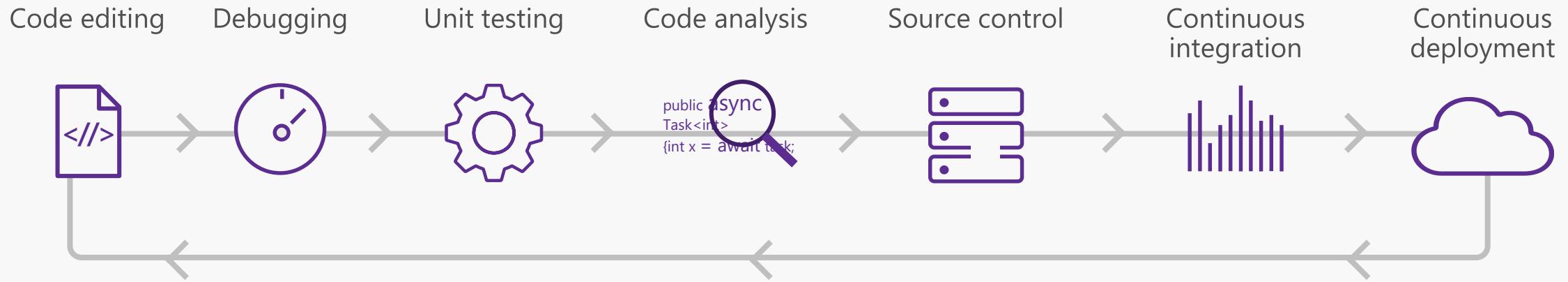


A traditional look at DevOps



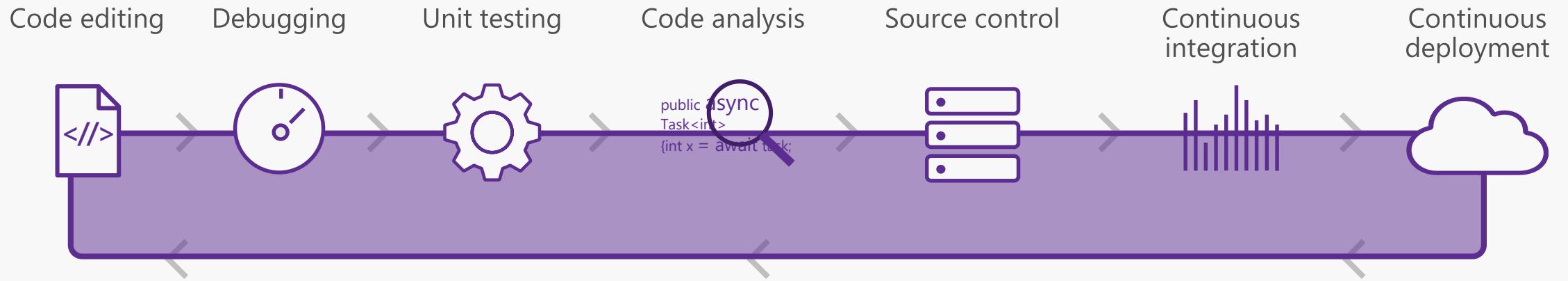
Development flows left to right and repeats

“Shift left”

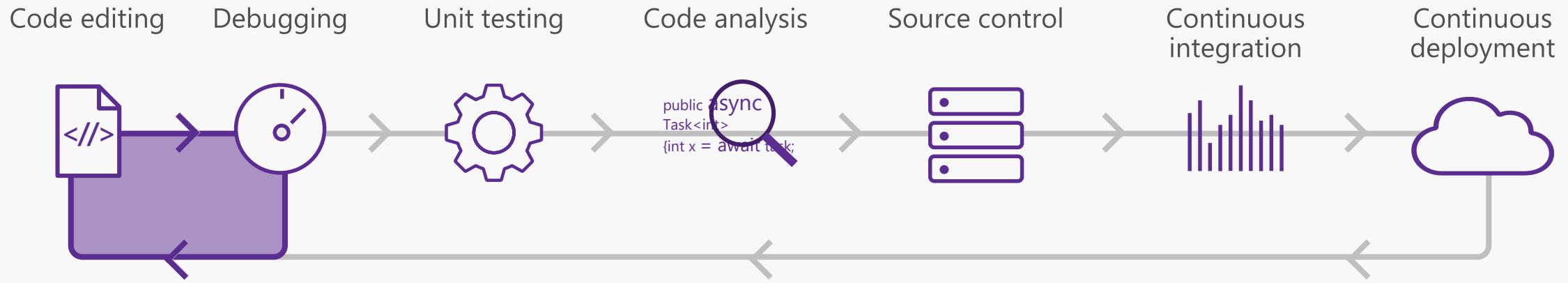


← Continuous integration + real-time experiences = **shift left**

“Shift left”

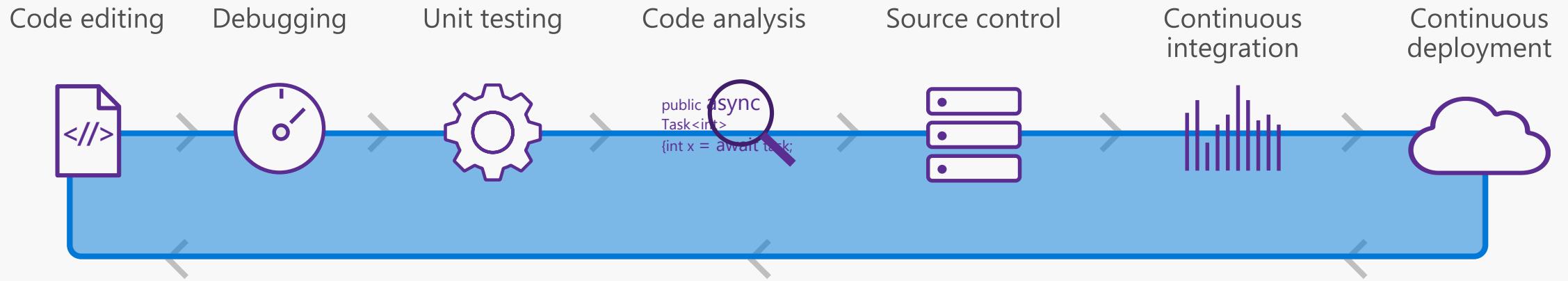


"Shift left"

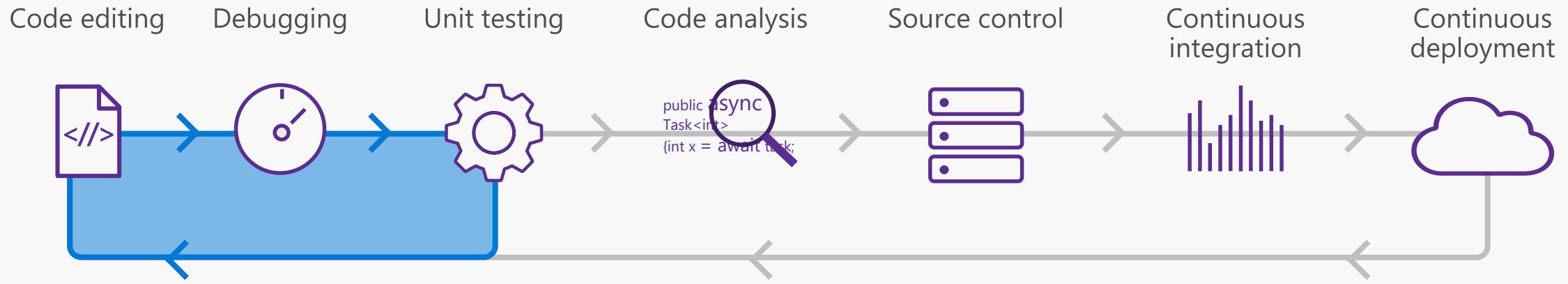


Edit and continue – pulled debugging into the edit/build cycle

“Shift left”

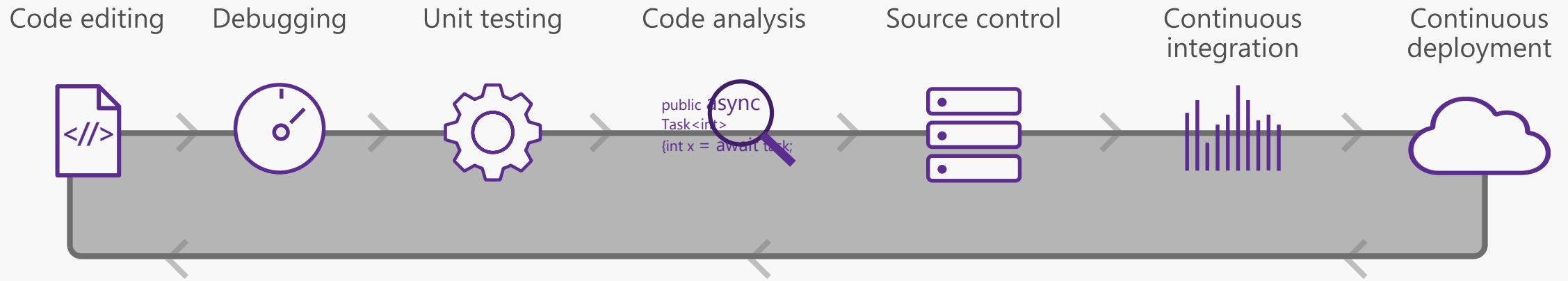


“Shift left”

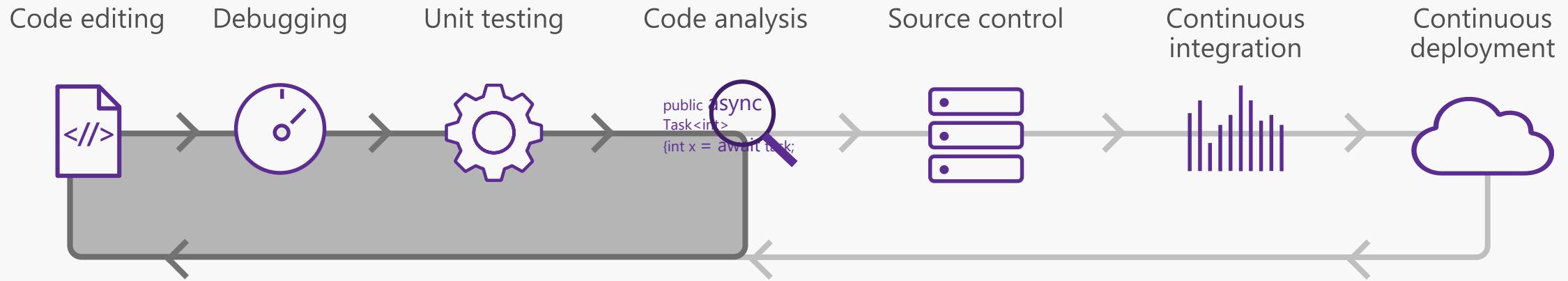


Live unit testing – pulls quality further into the inner loop

“Shift left”



“Shift left”



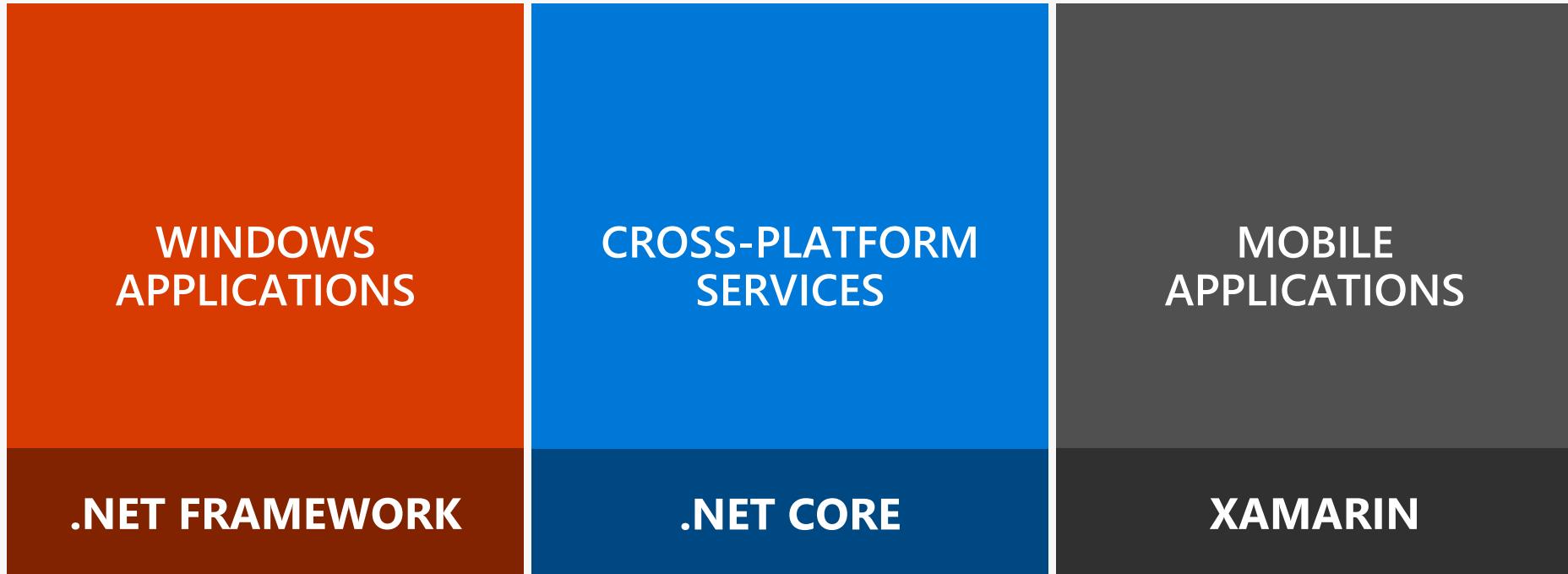
Live code analysis – immediate feedback loop

Demo

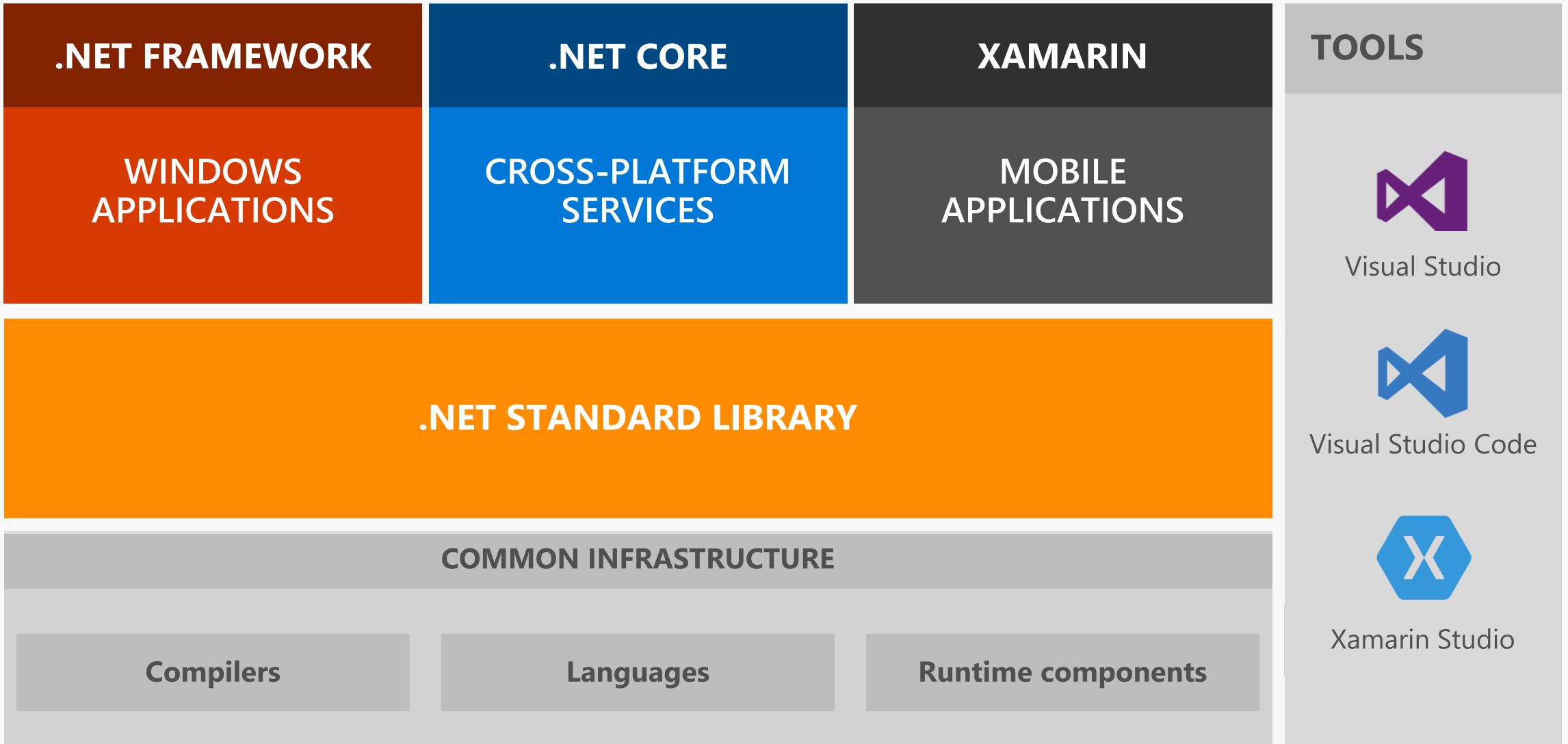


Photo used under CC <http://www.wocintechchat.com/>

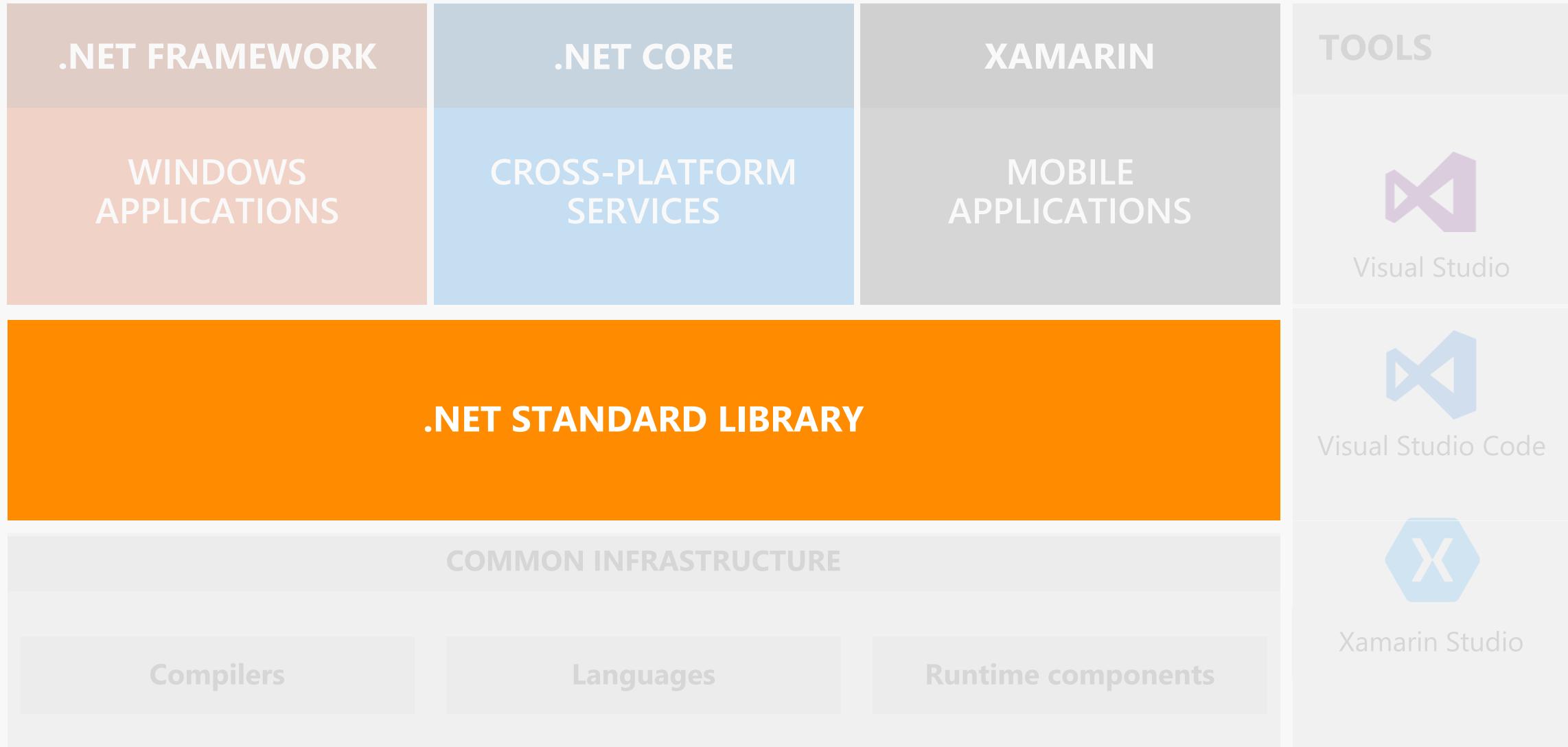
A new world for .NET



Unified platform



.NET Standard Library



.NET Standard Library

One library to rule them all

Learn one API

Reuse your code across .NET platforms

Easily consume third party / OSS libraries

.NET Standard 1.x

Evolution of Portable Class Libraries (PCLs) with a much simpler model

Two constraints:

- API surface relatively small, because .NET Core was small
- Can't reference the vast number of libraries already available for .NET Framework

.NET Standard 2.0

Much bigger API surface

Extended to cover intersection between .NET Framework and Xamarin

.NET Core will bring the new API surface to comply with .NET Standard 2.0

.NET Framework interop

Compatibility shim allows .NET Standard based libraries to reference existing .NET Framework binaries

No recompile required

Limited to .NET Framework libraries that only use APIs in .NET Standard (which are most of them)

APIs in .NET Standard 2.0

XML

XLinq • XML Document • XPath • Schema • XSL

SERIALIZATION

BinaryFormatter • Data Contract • XML

NETWORKING

Sockets • Http • Mail • WebSockets

IO

Files • Compression • MMF

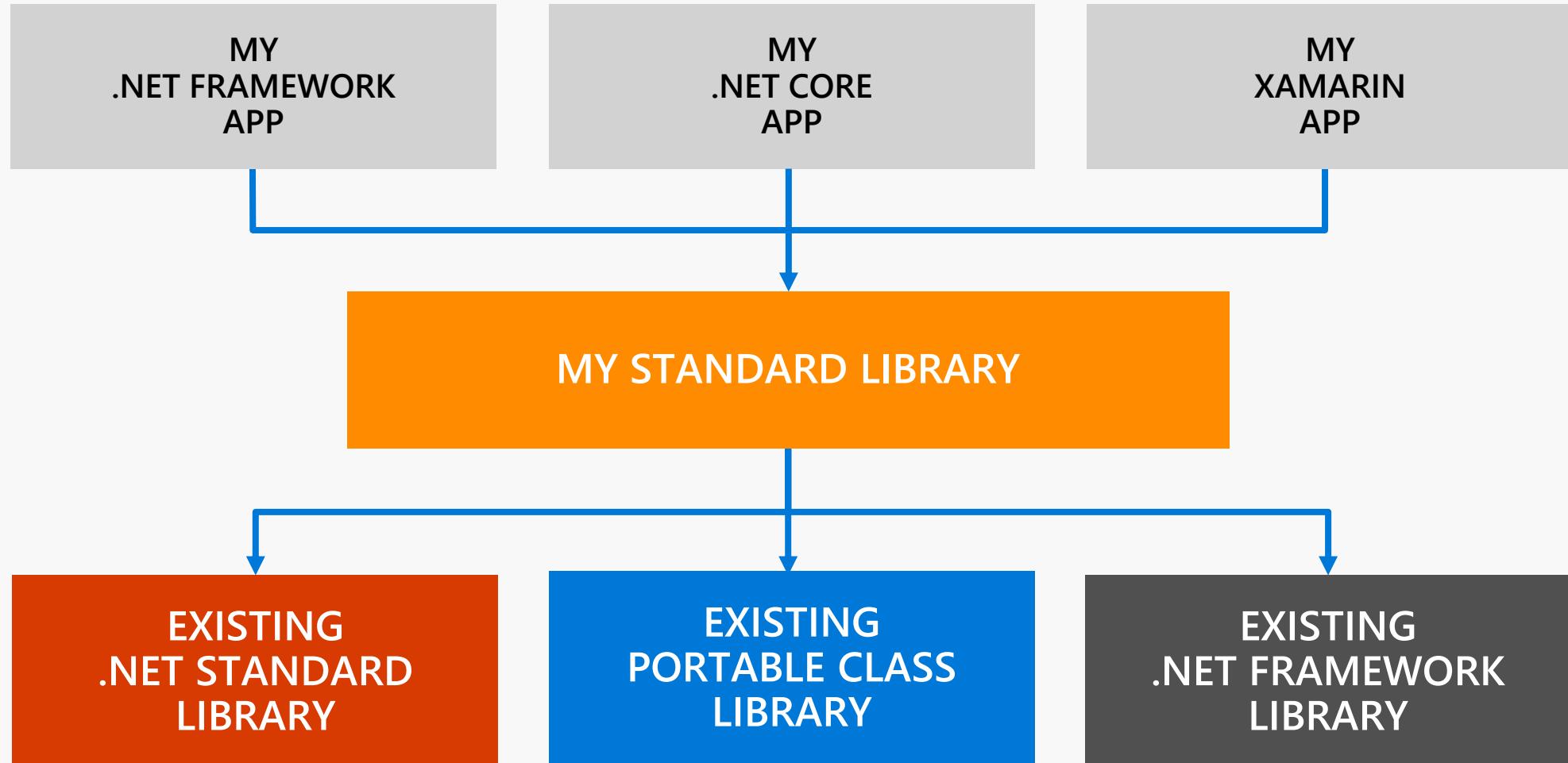
THREADING

Threads • Thread Pool • Tasks

CORE

Primitives • Collections • Reflection • Interop • Linq

.NET Standard 2.0 interop with .NET Framework



How does .NET Standard work?

.NET Standard is represented by:

The NuGet package **NetStandard.Library** which contains
The reference assembly **netstandard.dll**

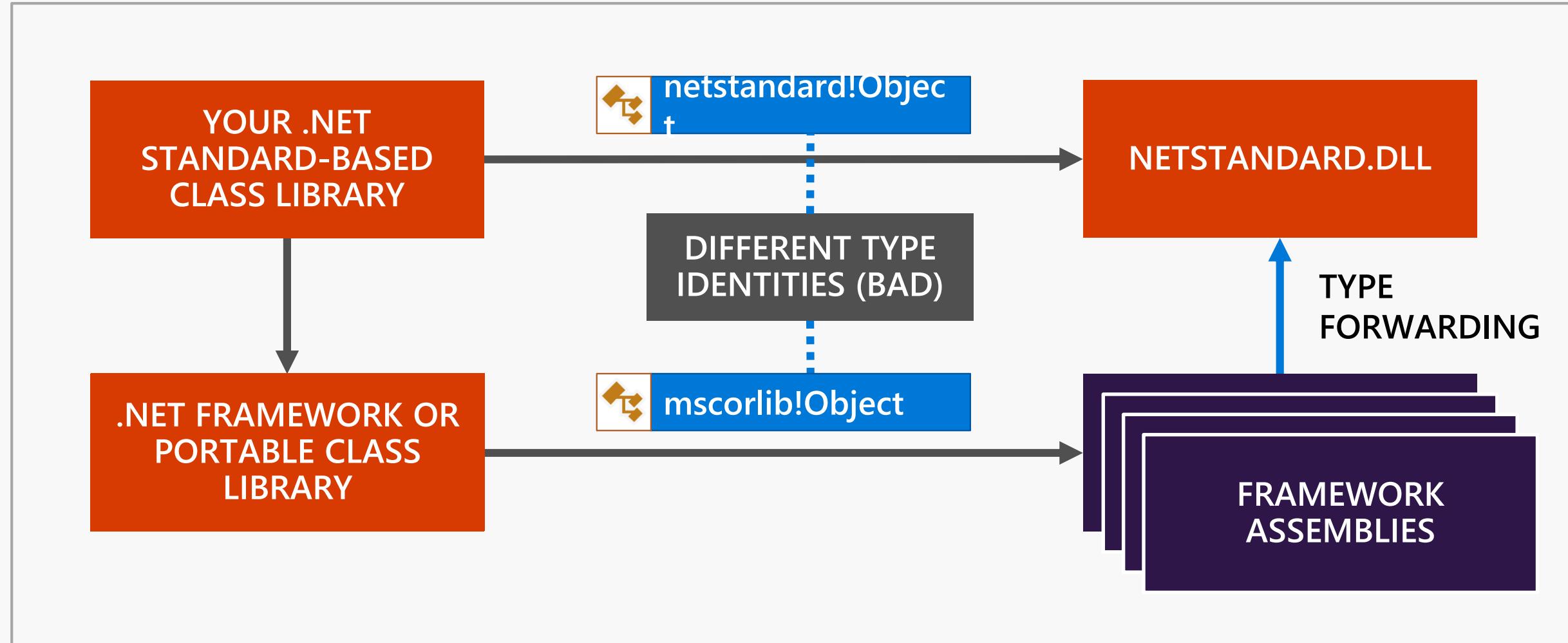
At build time

.NET Standard bridges references to existing .NET Framework and PCL assemblies via type forwarding

At runtime

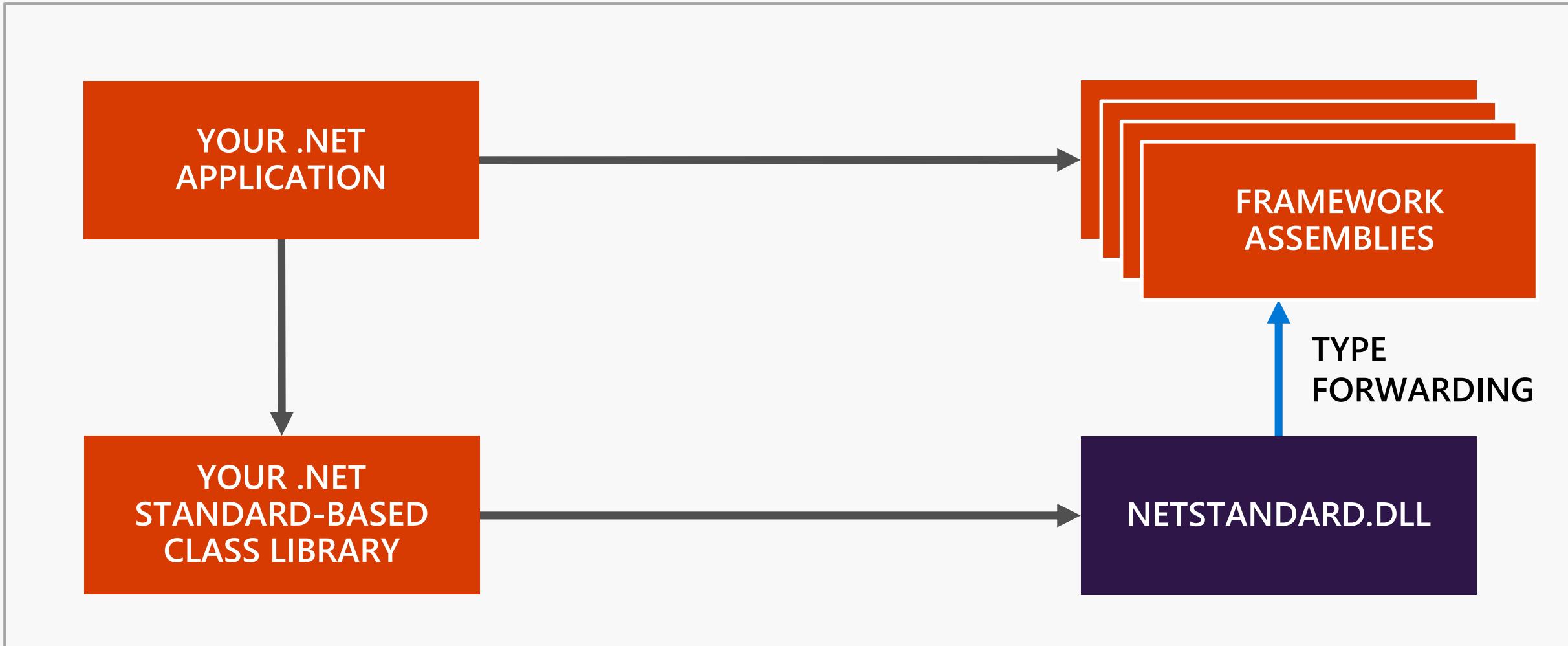
Each platform provides an implementation for netstandard.dll that type forwards to its implementation

.NET Standard under the hood



This happens when you build a .NET Standard-based Library

.NET Standard under the hood

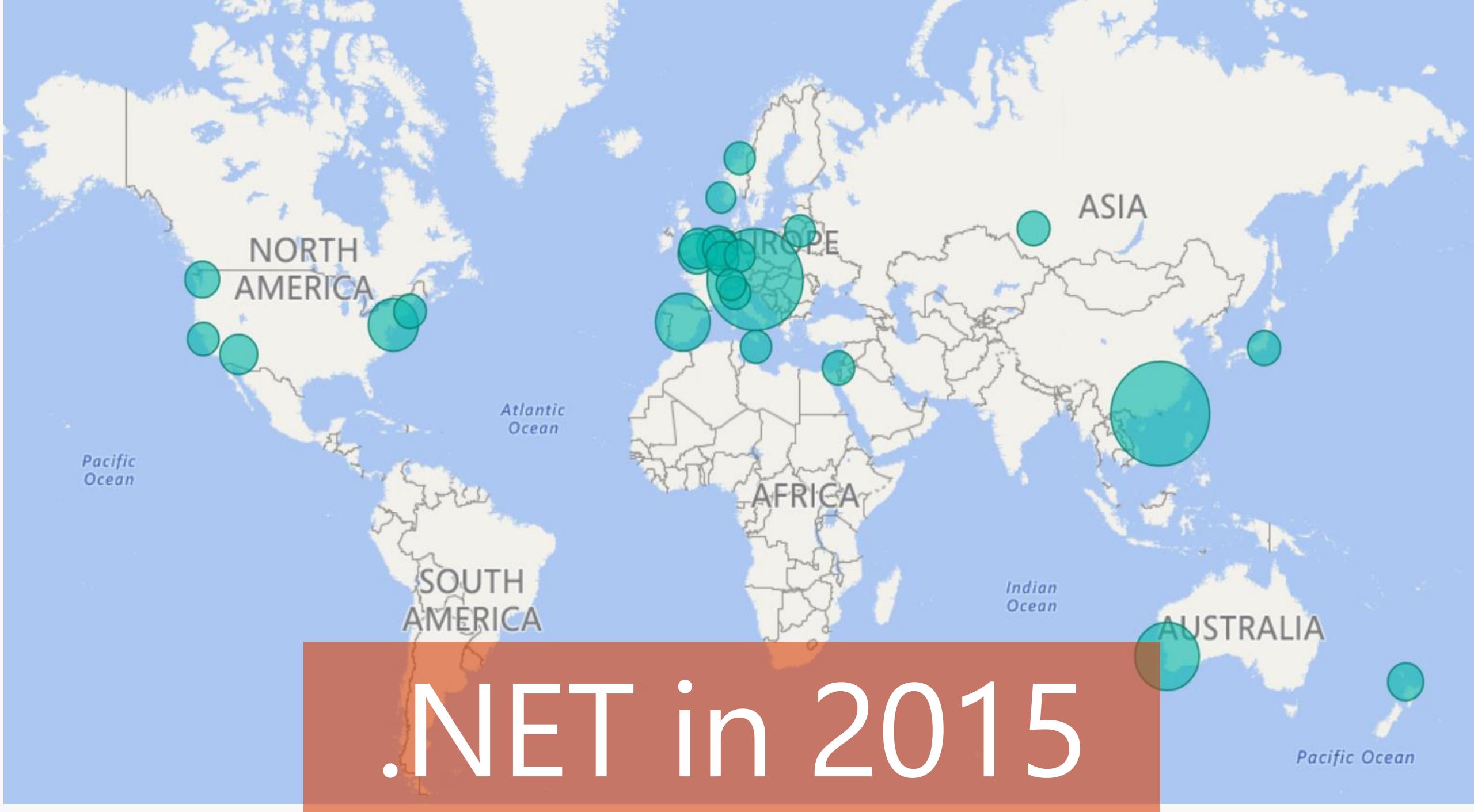


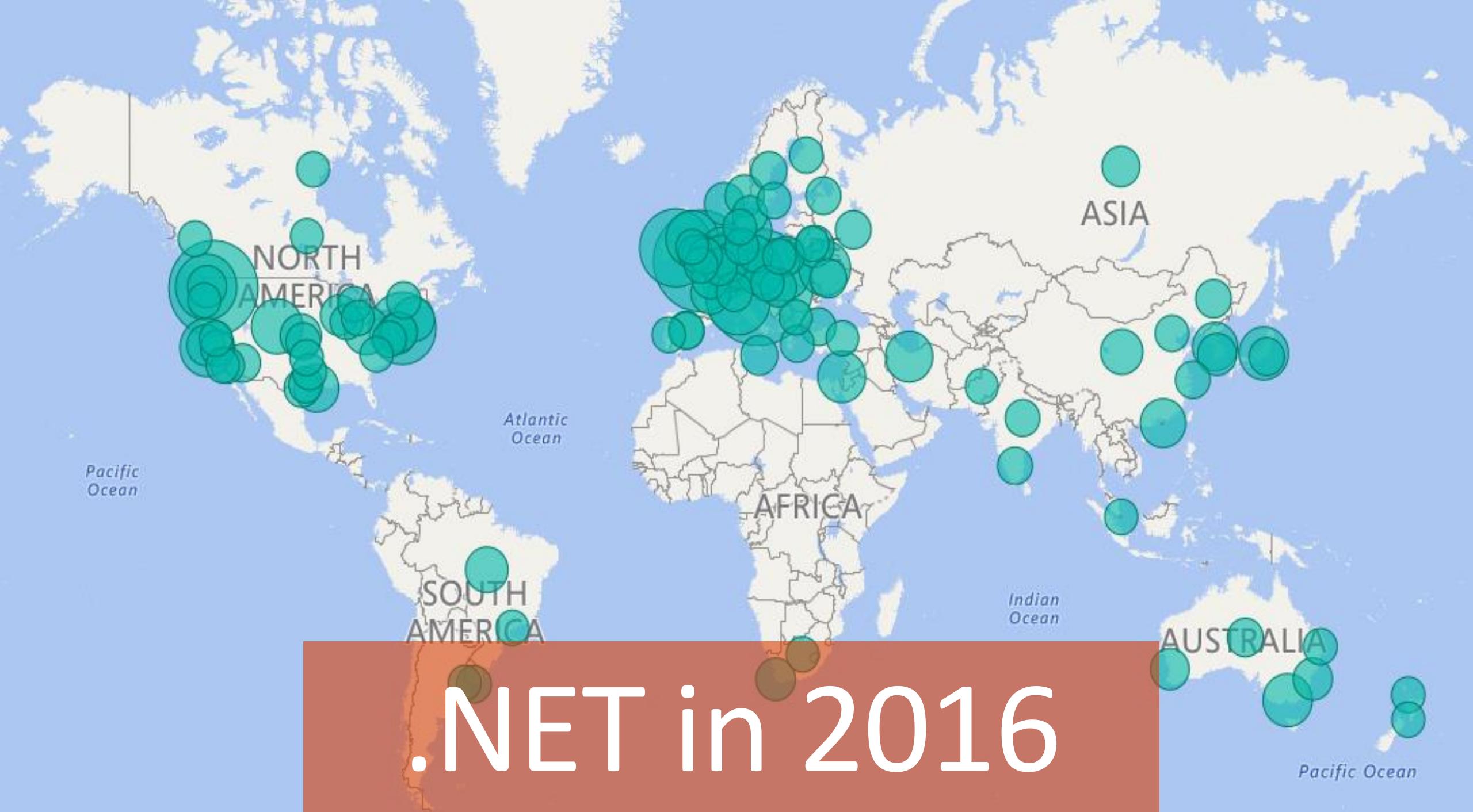
This happens when you load .NET Standard-library

Demo



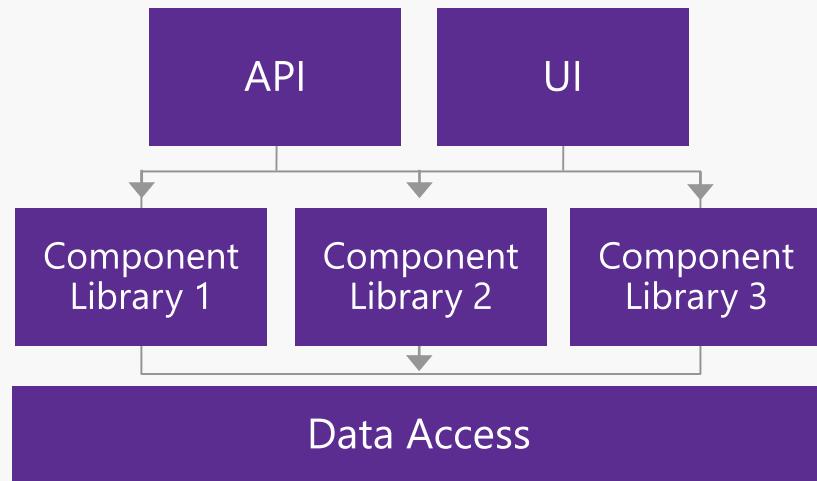
Photo used under CC <http://www.wocintechchat.com/>





Why microservices?

Traditional approach

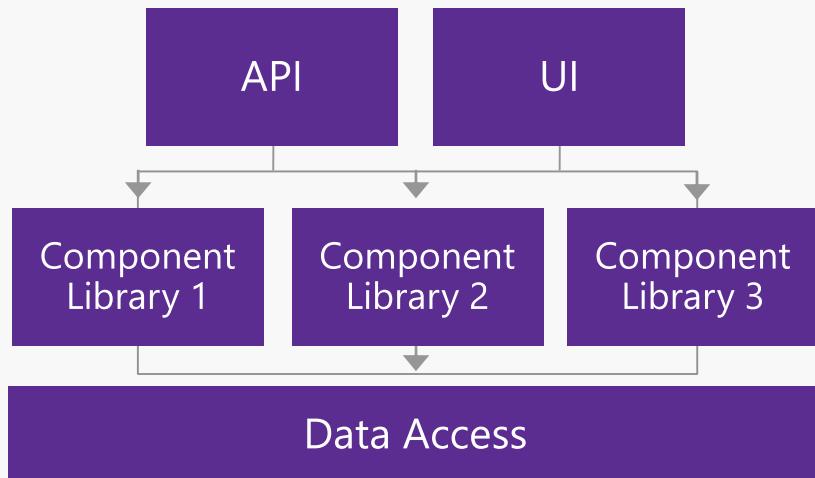


Microservices approach



Why microservices

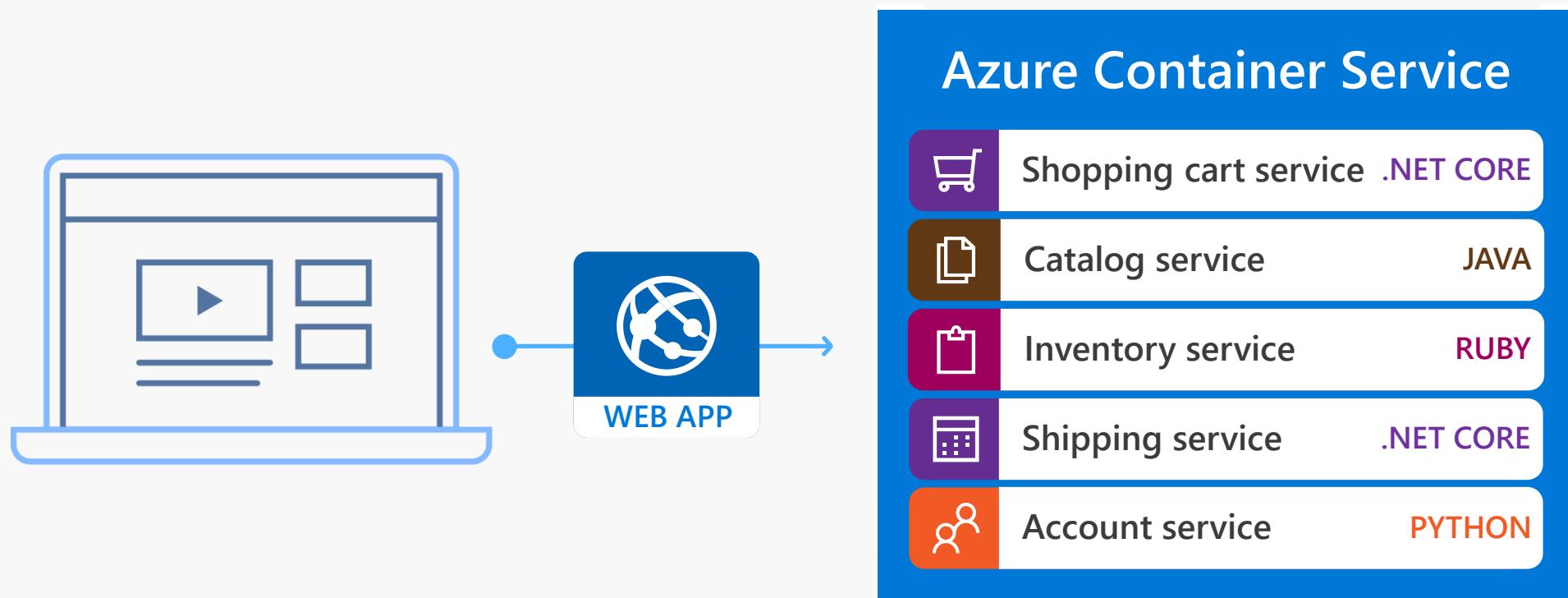
Traditional approach



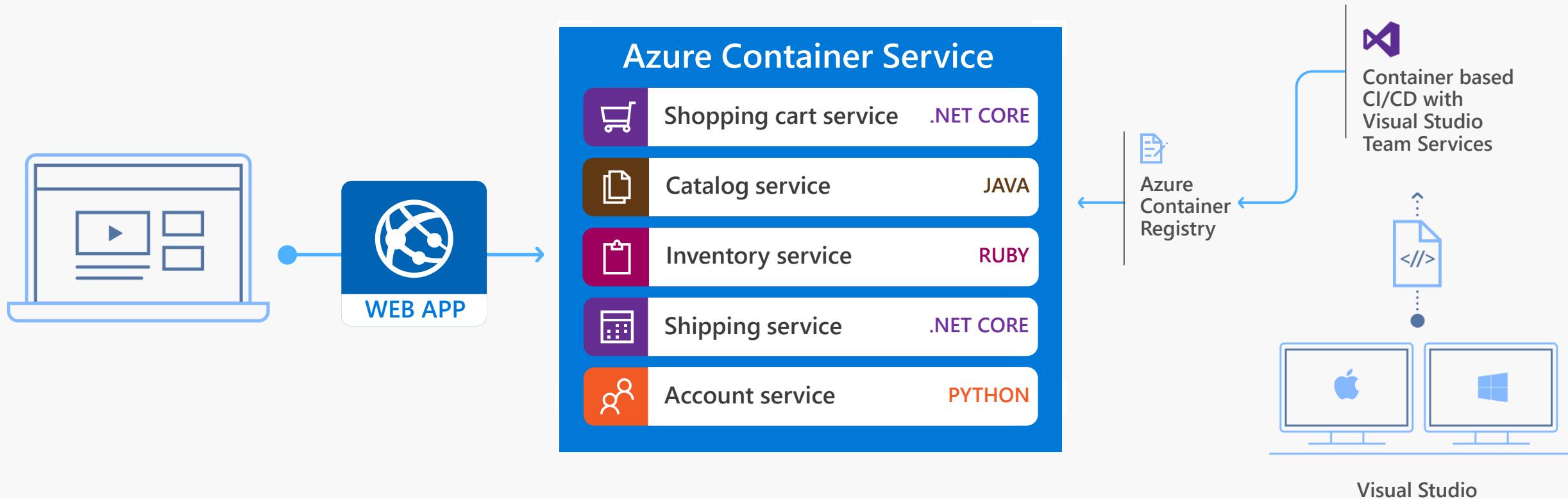
Microservices approach



Microservices + Containers



Continuous Integration with Containers





EXPAND MENU FOR MORE TOPICS



Windows/
Server

Active
Directory

Exchange

Cloud/Virt

SharePoint

Security

HOME

NEWS

FEATURES

PAPERS

WEBCASTS

SOFTWARE TRIALS

ADVERTISE

EVENTS

NEWSLETTERS

FREE SUBSCRIPTION

NEWS



2



Tweet



Share



44



Share 0

Windows Server 2016 Getting Linux Containers

By Scott Bekker ■ 04/20/2017

Microsoft this week announced that a new type of Linux container that runs on Windows Server 2016 will be coming soon.

The move, which was unveiled at this week's DockerCon 2017 in Austin, Texas, would break down a fundamental wall in deployment scenarios for containers to date. For now, Linux containers can only run on Linux host operating systems and Windows containers can only run on Windows



Most Popular Articles

OpenSSH Windows Nearing Completion

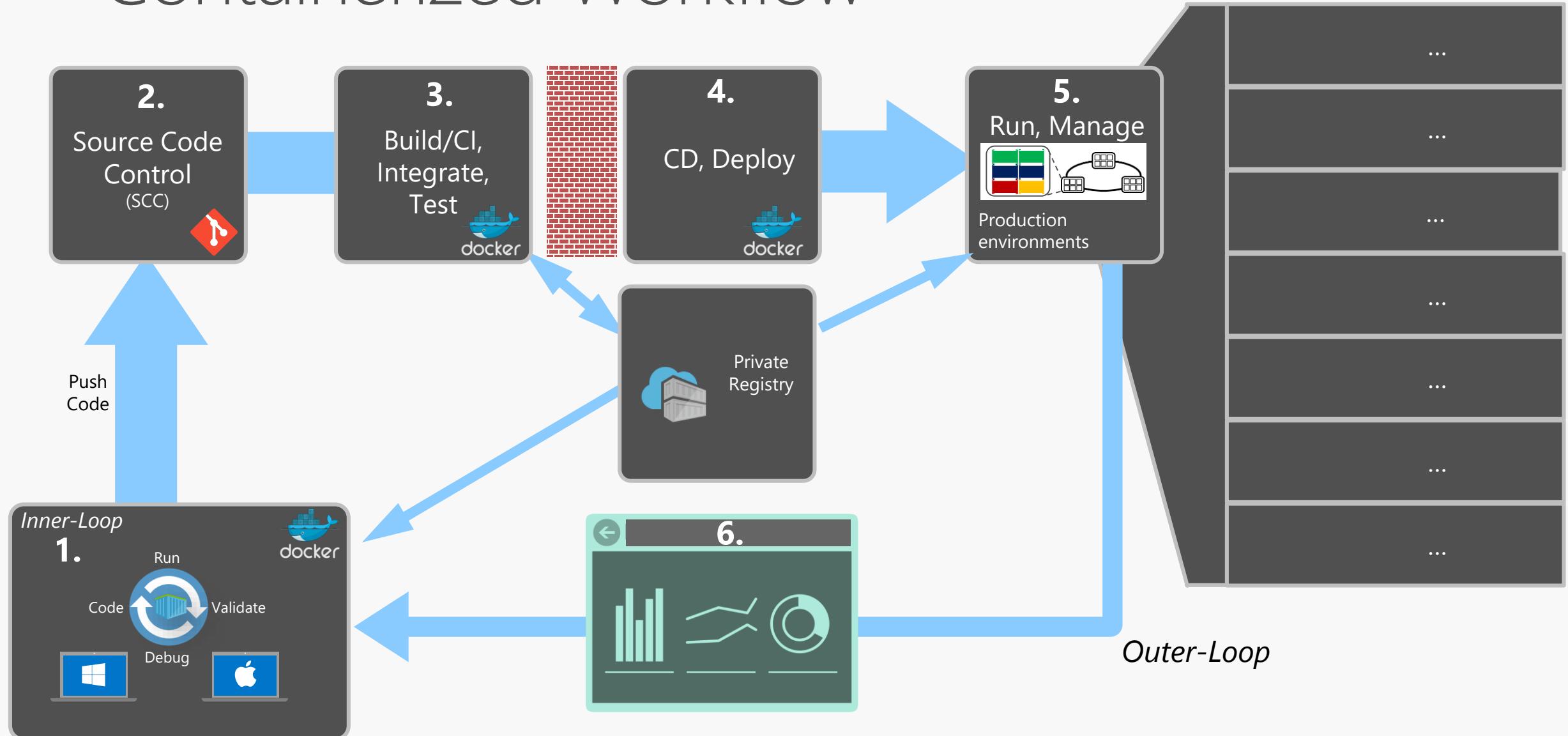
Does Everyone Still Need Windows?

Windows 10 Creators Update Tools and Documentation Released

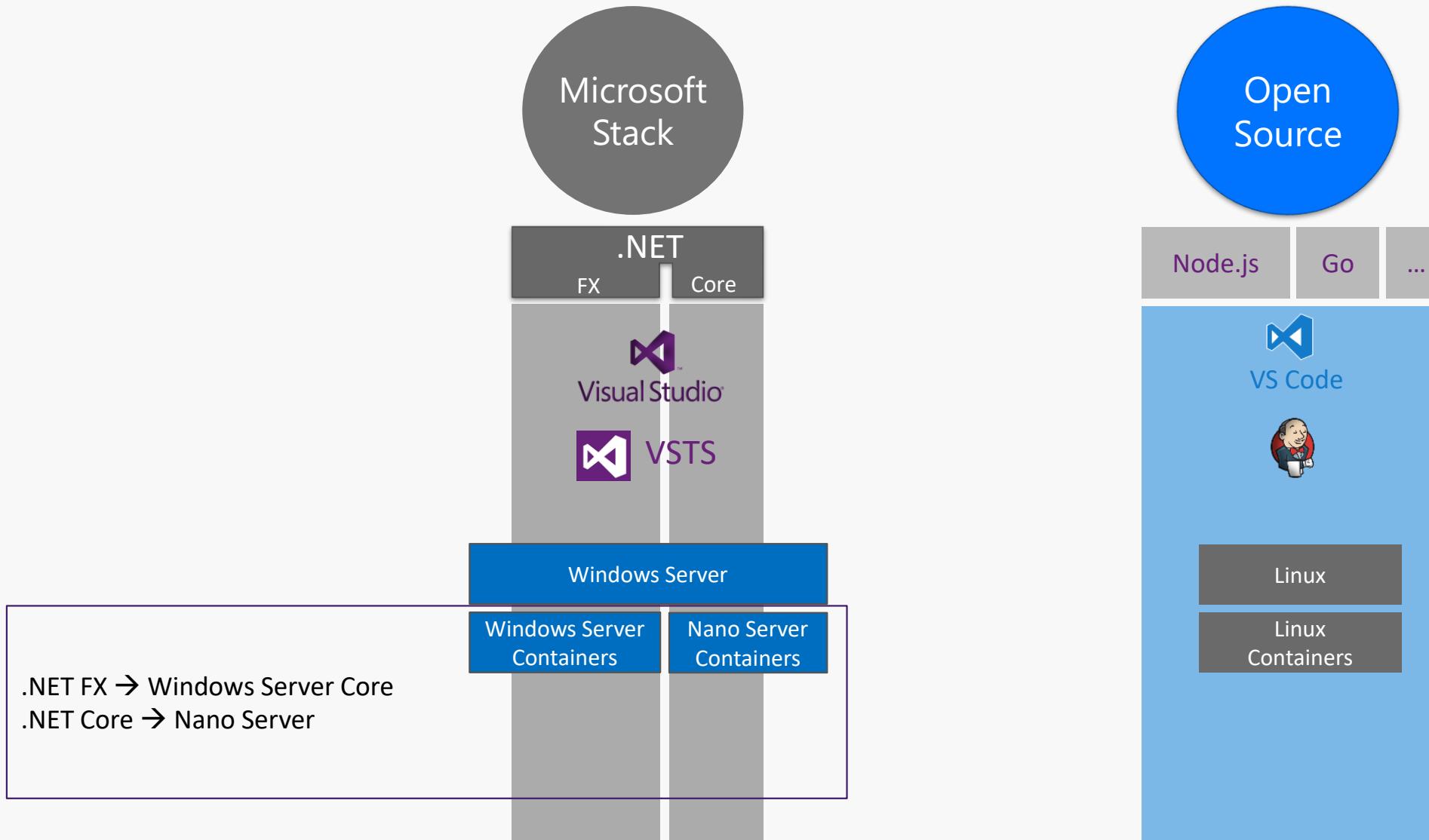
Microsoft Releases Windows 10 Creators Update and Security Patches

A Brave New World: Top Features Coming to SQL Server 2017

Containerized Workflow



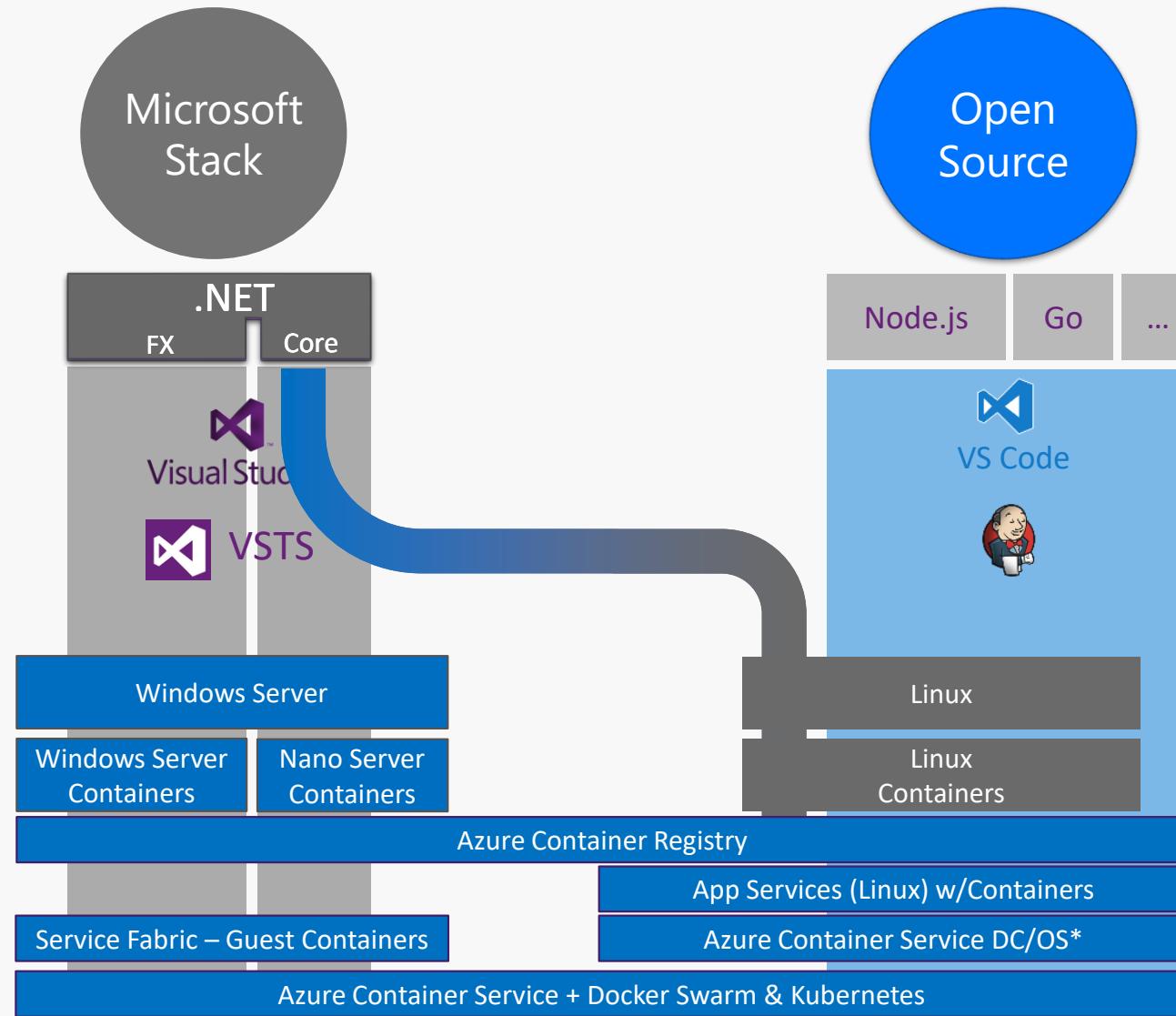
User Segments



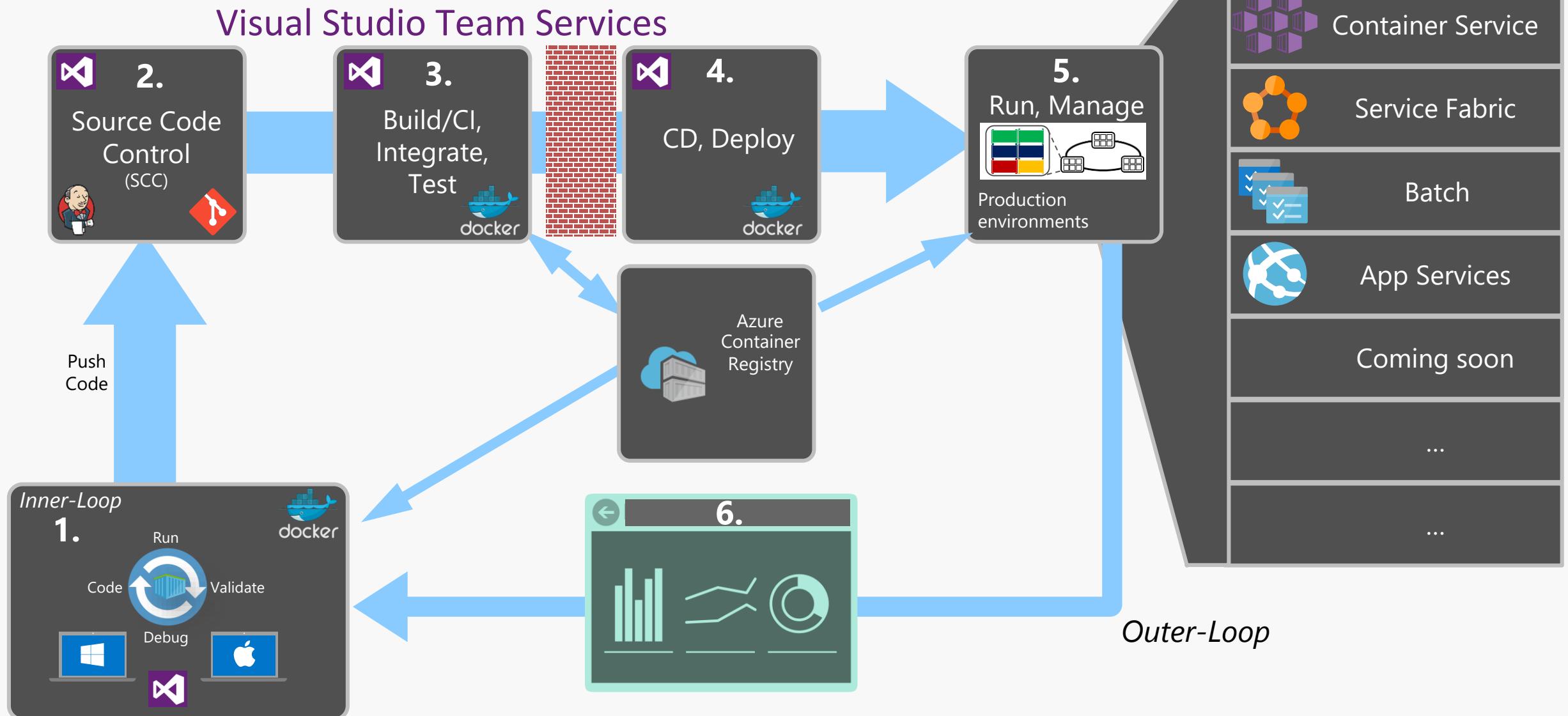
Visual Studio 2017

These are tooling paths

Customers will choose alternatives, that we will support from the platform, portal, docs and CLIs



Containerized Workflow



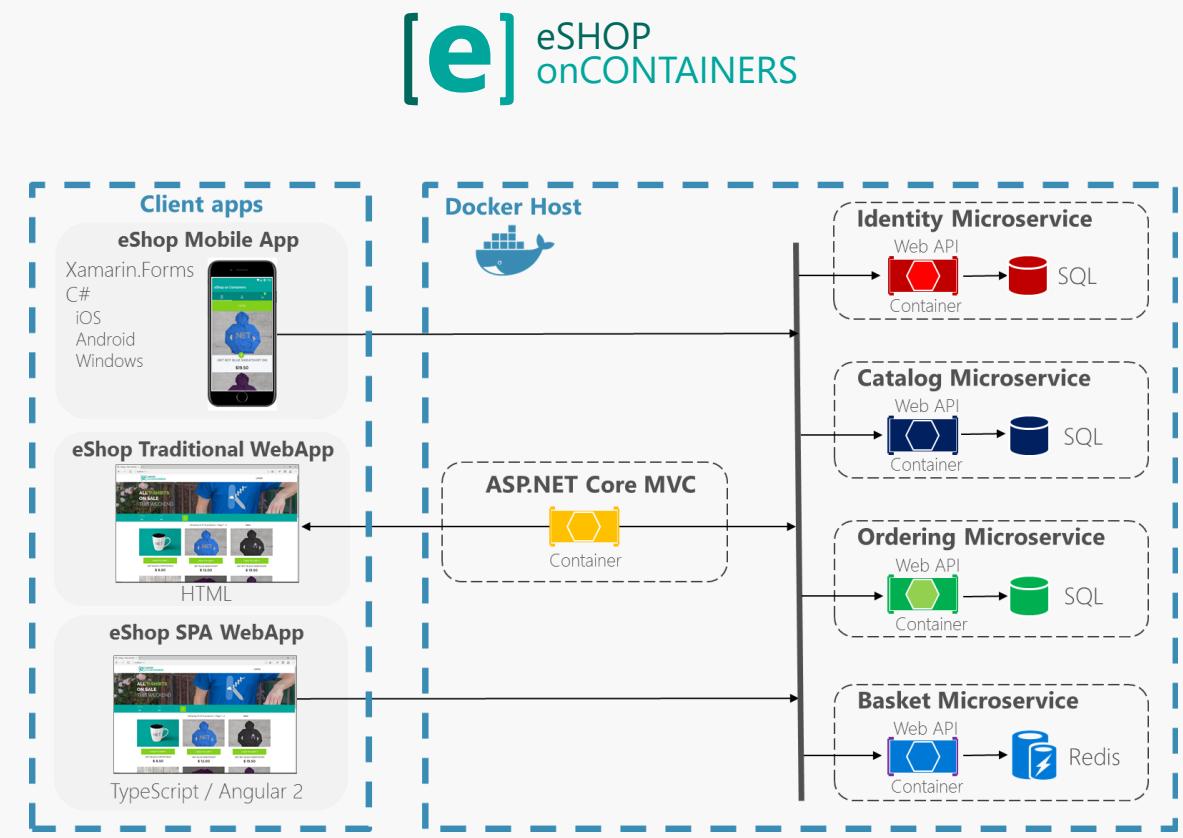
Demo



Photo used under CC <http://www.wocintechchat.com/>

.NET Core microservices reference app

- Includes backend services architected with best practices
- Mobile and web client apps included
- Supported in Visual Studio 2017
- Supported on Docker and dotnet CLI on Mac, Linux or Windows



Explore our beta release
and provide feedback:

<http://aka.ms/MicroservicesArchitecture>

Other VS2017 focus



DevOps



Mobile



Devops for Databases



VS Mobile Center



TFS import data



CI, testing, Monitoring



Docker integration



Xamarin tools



Test Impact Analysis



Inspector, Simulator



Microsoft
Innovation
University

Thank You!



RESOURCES

Fundamentals

github.com/Microsoft/VisualStudio2017Launch_KeynoteDemos

Containers

Azure.com/Tools

aka.ms/AzureContainerService

aka.ms/WindowsContainers

docs.microsoft.com/en-us/azure/container-service/container-service-setup-ci-cd

docs.microsoft.com/en-us/azure/container-service/container-service-connect