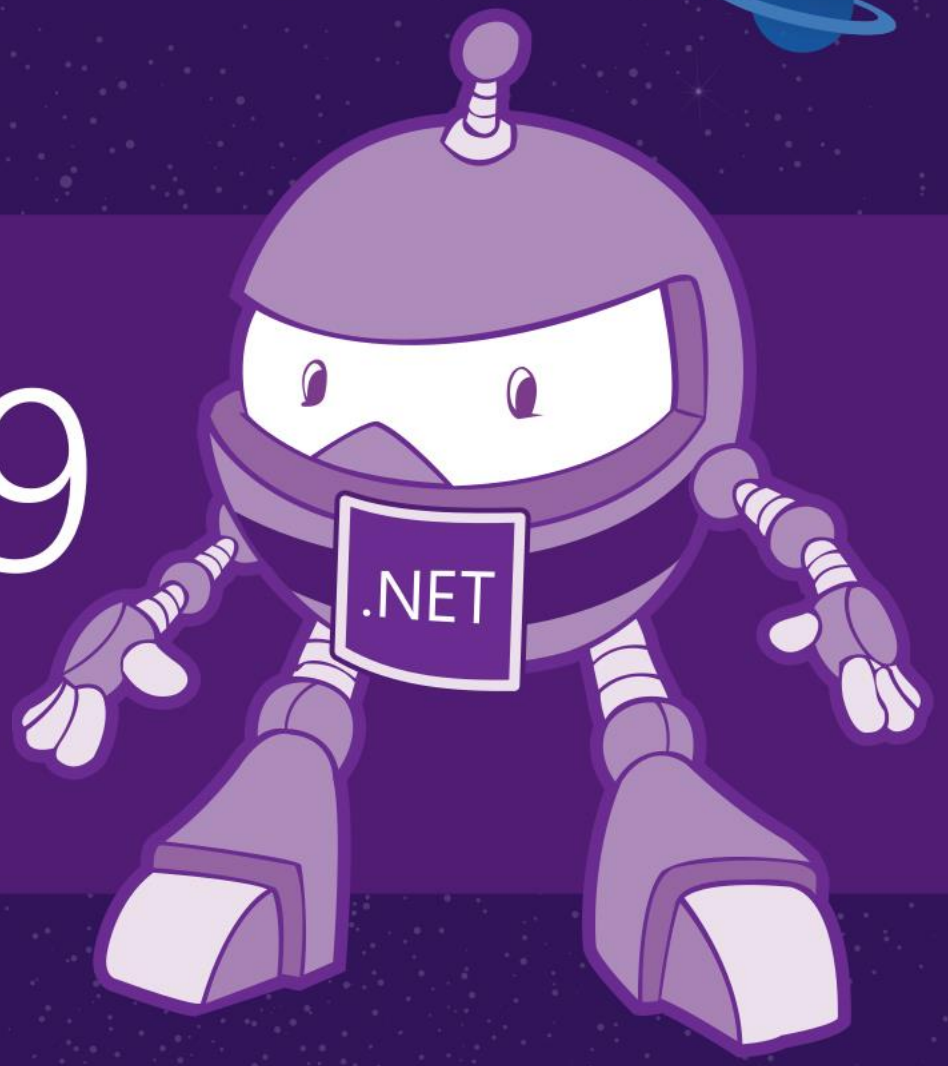
Microsoft

.NET Conf 2019

What's new in .NET Ecosystem | 10 Ottobre 2019



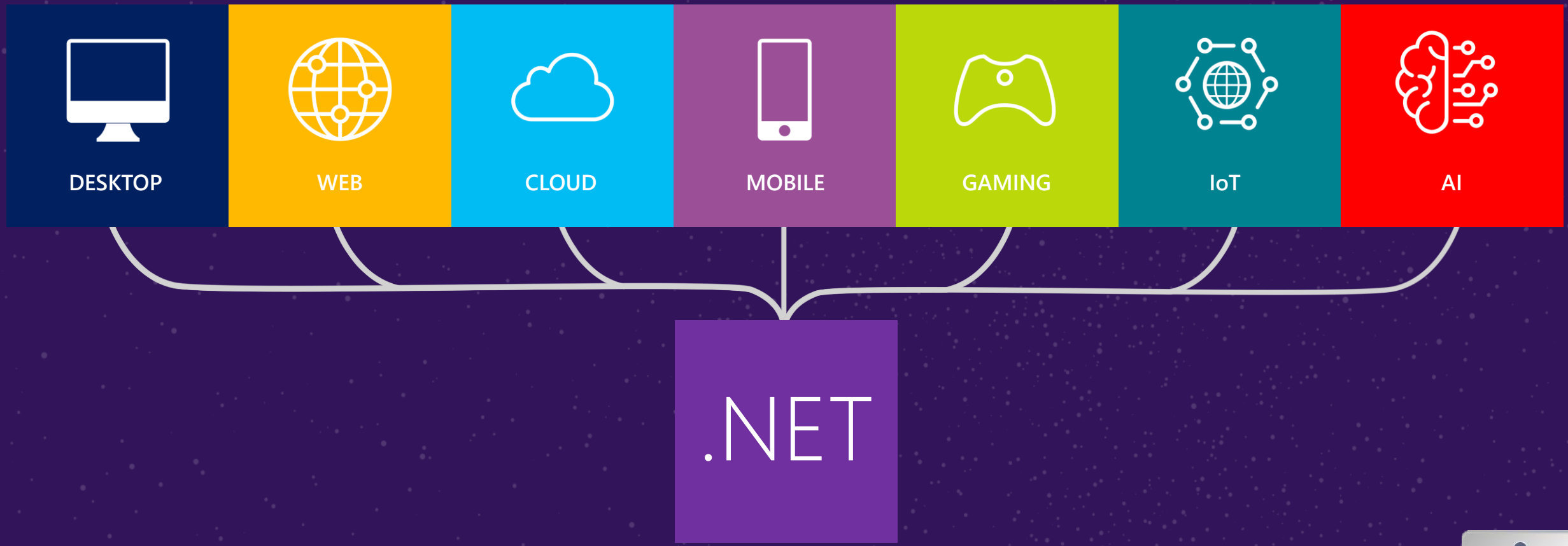
MVP
Microsoft® Most Valuable Professional

Clemente Giorio @tinux80

Gianni Rosa Gallina @giannirg

www.dotnetconf.net

Your platform for building **anything**



RELEASED

.NET Core 3.0

WPF and Windows Forms support

Side-by-side support & self-contained EXEs

Full-stack web development with C# and Razor

New C# 8.0 language features

dot.net/get-core3

RELEASED

Visual Studio 2019 version 16.3
Visual Studio 2019 for Mac version 8.3

Support for .NET Core 3.0 and C# 8.0

New .NET productivity improvements

More performance improvements

iOS 13 & Android 10 support

visualstudio.com/downloads

C# 8.0

.NET

Torino Technologies Group

C# 8.0

Safe

Nullable and non-nullable reference types help you write safer code

Declare your intent more clearly

Modern

Async streams for modern workloads like cloud & IoT communication

Easily work with cloud scale datasets using indexes and ranges

Productive

Write less code using patterns

Protect data with readonly members

Improved using statements for resource management

C# 8.0

Nullable reference types

References must be declared nullable, to be null-assigned

```
string? s = null;
```

```
<Nullable>enable</Nullable>
```

DEMO

C# 8.0

Torino Technologies Group

Library authors

“Nullable adoption phase”

- Try to get “nullified” before .NET 5

Consider multi-targeting

- .NET Core 3.0 for nullable annotations
- .NET Standard 2.0 for reach
- Manually opt in to C# 8.0 *(not supported)*

C# 8.0

Async streams

Introduce **IAsyncEnumerable<T>**, an asynchronous version of IEnumerable<T>, with `await foreach` and `yield return` support
https://www.infoq.com/articles/Async-Streams

Ranges, Indices and Recursive Patterns

New index '^' and range '..' operators, similar to Python behavior
https://www.infoq.com/articles/cs8-ranges-and-recursive-patterns

Default implementations of interface members

https://www.infoq.com/articles/default-interface-methods-cs8

C# 8.0

Async streams

Introduce **IAsyncEnumerable<T>**, an asynchronous version of IEnumerable<T>, with `await foreach` and `yield return` support

https://www.infoq.com/articles/Async-Streams

```
public static decimal CalculateToll(object vehicle) =>

    vehicle switch
    {
        Car _ => 2.00m,
        Taxi _ => 3.50m,
        Bus _ => 5.00m,
        DeliveryTruck _ => 10.00m,
        { } => throw new ArgumentException(message: "Not a known vehicle type", paramName:
        null => throw new ArgumentNullException(nameof(vehicle))
    };
```

Torino Technologies Group

C# 8.0

Async streams

Introduce **IAsyncEnumerable<T>**, an asynchronous version of IEnumerable<T>, with `await foreach` and `yield return` support

https://www.infoq.com/articles/Async-Streams

```
public static decimal CalculateToll(object vehicle) =>

    vehicle switch
    {
        Car { Passengers: 3} => 1.0m,
        Car {Passengers:2 } => 1.50m,
        Car _ => 2.0m,
        Taxi _ => 3.50m,
        Bus _ => 5.00m,
        DeliveryTruck _ => 10.00m,
        { } => throw new ArgumentException(message: "Not a known vehicle type", paramName:
        null => throw new ArgumentNullException(nameof(vehicle))
    };
```

Torino Technologies Group

C# 8.0

Async streams

Introduce **IAsyncEnumerable<T>**, an asynchronous version of IEnumerable<T>, with `await foreach` and `yield return` support

https://www.infoq.com/articles/Async-Streams

```
public decimal PeakTimePremium(DateTime timeOfToll, bool inbound) =>
            (IsWeekDay(timeOfToll), GetTimeBand(timeOfToll), inbound) switch
            {
                (true, TimeBand.Overnight, _) => 0.75m,
                (true, TimeBand.Daytime, _) => 1.5m,
                (true, TimeBand.MorningRush, true) => 2.0m,
                (true, TimeBand.EveningRush, false) => 2.0m,
                (_, _, _) => 1.0m,
            };
```

C# 8.0

Async strea

Introduce

IEnumera ort

https://www

Ranges, Ind

New index '^' a

https://www.info

Default imp

https://www.info

```
private static string[] words = {
                    // index from start    index from end
    "The",          // 0                   ^9
    "quick",        // 1                   ^8
    "brown",        // 2                   ^7
    "fox",          // 3                   ^6
    "jumped",       // 4                   ^5
    "over",         // 5                   ^4
    "the",          // 6                   ^3
    "lazy",         // 7                   ^2
    "dog"           // 8                   ^1
};

public static void Demo1()
{
    Console.WriteLine(words[^1]);
}

public static void Demo2()
{
    var lazyDog = words[^2..^0];
    foreach (var word in lazyDog) Console.Write($"< {word} >");
    Console.WriteLine();
}
```

Torino Technologies Group

C# 8.0

Async streams

Introduce **IAsyncEnumerable<T>**, an asynchronous version of IEnumerable<T>, with `await foreach` and `yield return` support
https://www.infoq.com/articles/Async-Streams

Ranges, Indices and Recursive Patterns

New index '^' and range '..' operators, similar to Python behavior
https://www.infoq.com/articles/cs8-ranges-and-recursive-patterns

Default implementations of interface members

https://www.infoq.com/articles/default-interface-methods-cs8

https://docs.microsoft.com/en-us/dotnet/csharp/whats-new/csharp-8

Torino Technologies Group

C# 8.0

Async streams

Introduce **IAsyncEnumerable<T>**, an asynchronous version of IEnumerable<T>, with `await foreach` and `yield return` support

h

Ran

New

https

Def

https

```
interface IDefaultInterfaceMethod
 {
   public void DefaultMethod()
   {
     Console.WriteLine("I am a default method in the interface!");
   }
 }

 class AnyClass : IDefaultInterfaceMethod
 {
 }
```

https://docs.microsoft.com/en-us/dotnet/csharp/whats-new/csharp-8

C# 8.0

Blog posts:

Try out Nullable Reference Types: devblogs.microsoft.com/dotnet/try-out-nullable-reference-types
Take C# 8.0 for a spin: devblogs.microsoft.com/dotnet/take-c-8-0-for-a-spin
Nullable Reference Types in C#: devblogs.microsoft.com/dotnet/nullable-reference-types-in-csharp

.NET Conf Sessions

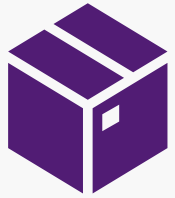
What's new in C# 8.0, Part 1 and 2
https://www.youtube.com/watch?v=TJiLhRPgyq4&list=PLReL099Y5nRd04p81Q7p5TtyjCrj9tz1t&index=3
https://www.youtube.com/watch?v=fhf8N4004u0&list=PLReL099Y5nRd04p81Q7p5TtyjCrj9tz1t&index=4
Introduction to Async Stream in the Real World
https://www.youtube.com/watch?v=Ylcl8hKks_Y&list=PLReL099Y5nRd04p81Q7p5TtyjCrj9tz1t&index=56

Windows desktop apps

.NET

Torino Technologies Group

.NET Core 3.0 for Windows Desktop



Deployment Flexibility

Side-by-side deployment, self-contained EXEs

Install machine global or app local framework

Windows 10

Access modern Windows 10 APIs from WPF and WinForms

Use native Windows 10 controls via XAML islands

Open Source

WPF and WinForms projects also open source on GitHub

Take advantage of performance, runtime and API improvements happening in .NET Core

RELEASED

App Center for .NET Core 3.0 Windows Apps

Easily deploy your app to your QA teams, testers, and app stores

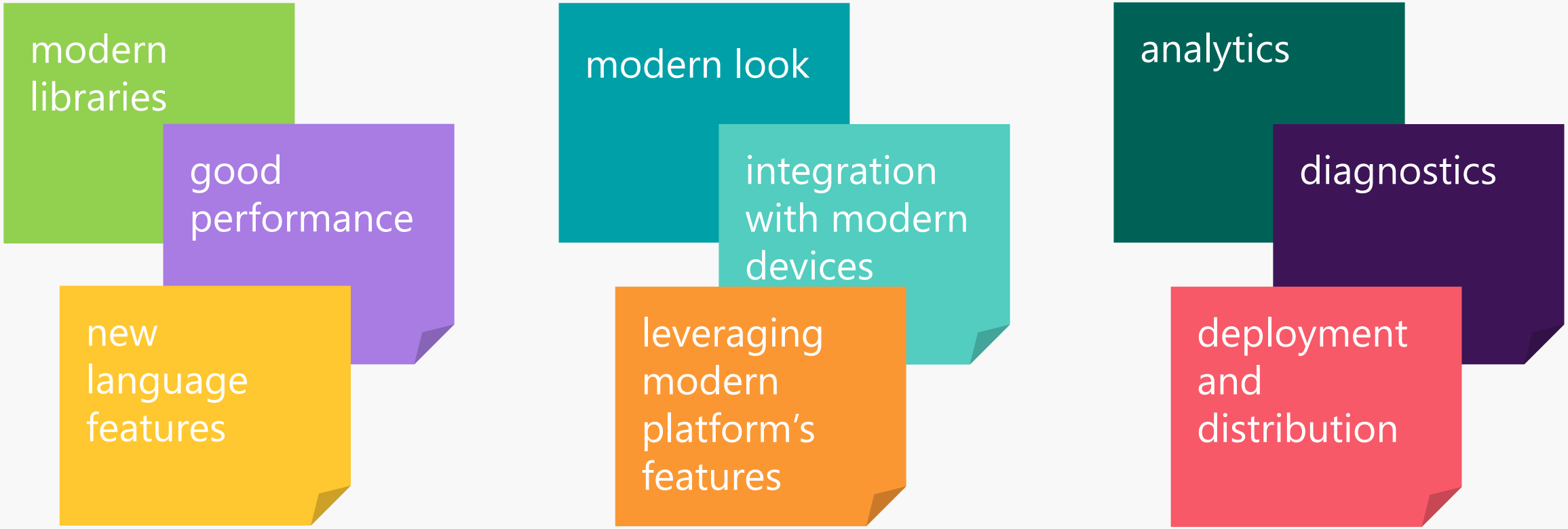
Collect real-time app diagnostics data, prioritize and fix issues faster

Track usage patterns, user adoption, and other engagement metrics

Now supports WPF and WinForms apps on .NET Core 3.0

Appcenter.ms

Upgrades for Windows desktop development



Why .NET Core for desktop

- .NET Core is the future for .NET
- .NET Core-specific features
  - Side-by-side deployment of different .NET Core versions
  - Self-contained applications
  - Single-file executables
  - Smaller app sizes
  - …and many more

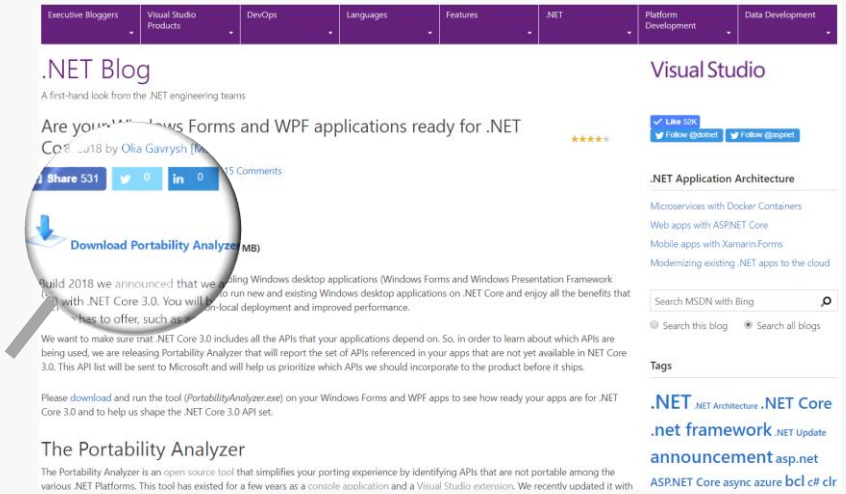
How to choose: .NET Framework or .NET Core?

- New apps
  - .NET Core
- Completed apps in maintenance
  - OK to leave on .NET Framework
- Existing apps in active development
  - Consider porting to .NET Core

How compatible is my app with .NET Core?

Portability Analyzer

- will show all APIs from your code that are not in .NET Core 3.0
- https://aka.ms/portabilityAnalyzer



Torino Technologies Group

Porting

Porting to .NET Core\Standard is done by updating

```
<Project Sdk="Microsoft.NET.Sdk">
  <PropertyGroup>
    <TargetFramework>net472</TargetFramework>
```

```
<Project Sdk="Microsoft.NET.Sdk">
  <PropertyGroup>
    <TargetFramework>netcoreapp3.0</TargetFramework>
```

If your .csproj file has old style, first migrate it to the new SDK-style

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

  <PropertyGroup>
    <OutputType>WinExe</OutputType>
    <TargetFramework>netcoreapp3.0</TargetFramework>
    <UseWinForms>true</UseWinForms>
    <GenerateAssemblyInfo>false</GenerateAssemblyInfo>
  </PropertyGroup>

</Project>
```

Torino Technologies Group

Porting by hand

1. If you have **packages.config**, right-click & “Migrate packages.config to PackageReference”
2. Update content of .csproj file with a .csproj file from blank .NET Core project
3. Add the references and resources from the old .csproj file:
   - <PackageReference>
   - <ProjectReference>
   - <Content>, <Resource>,…

Porting with Try Convert

Run from command line:

```
.\try-convert.exe -p "<path to your .csproj file>"
```

Windows Compatibility Pack

NuGet package Microsoft.Windows.Compatibility

- Can be referenced from .NET Core & .NET Standard
- Has ~21k APIs (Windows-only as well as cross-platform)

Contents

ACLs
Code Pages
CodeDom
Configuration
Crypto
DirectoryServices
EventLog
MEF
Odbc
Perf Counters
Permissions
...
Ports
Registry
Runtime Caching
WCF
Windows Services

DEMO

.NET Core 3.0
WPF & Windows Forms

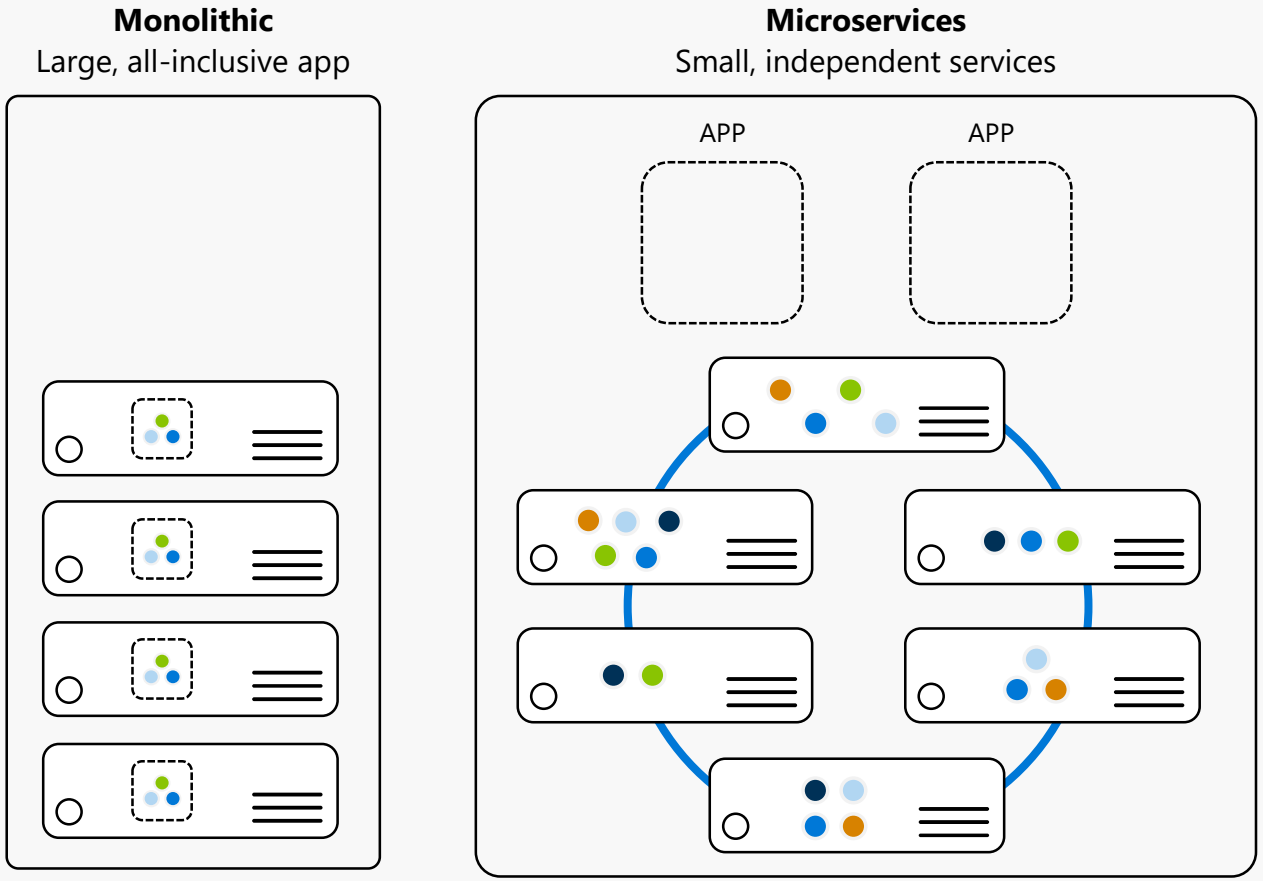
Microservices

.NET

Torino Technologies Group

Microservices: for faster app development

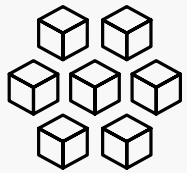
- Independent deployments
- Improved scale and resource utilization per service
- Smaller, focused teams



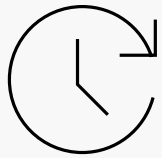
Torino Technologies Group

Azure Kubernetes Service (AKS)

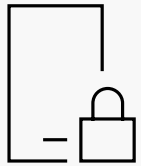
Ship faster, operate easily, and scale confidently with managed Kubernetes on Azure



Manage Kubernetes with ease



Accelerate containerized development



Build on an enterprise-grade, secure foundation

Run anything, anywhere

ASP.NET Core 3.0

gRPC

High performance contract-based RPC services with .NET

Works across many languages and platforms

Worker Service

Starting point for long running back processes like Windows Server or Linux daemon

Producing or consuming messages from a message queue

Web API's + Identity

Add security and authentication to Web API's

Sessions

.NET Core 3.0 Microservices

.NET Conf 2019 Keynote
https://youtu.be/1xQE2bWkwjo?list=PLReL099Y5nRd04p81Q7p5TtyjCrj9tz1t&t=294

Building Cloud Native Apps with .NET Core 3.0 and Kubernetes
https://www.youtube.com/watch?v=A0i5BUoKu6s&list=PLReL099Y5nRd04p81Q7p5TtyjCrj9tz1t&index=9

What's new in SignalR with .NET Core 3.0
https://www.youtube.com/watch?v=dHiETzo6GB8&list=PLReL099Y5nRd04p81Q7p5TtyjCrj9tz1t&index=11

Architecting .NET Microservices in a Docker Ecosystem
https://www.youtube.com/watch?v=ymcLM-TNk1o&list=PLReL099Y5nRd04p81Q7p5TtyjCrj9tz1t&index=68

Putting the ".NET" into "Kubernetes"
https://www.youtube.com/watch?v=GBOPBfcJ2zM&list=PLReL099Y5nRd04p81Q7p5TtyjCrj9tz1t&index=66

Mobile apps with Xamarin

.NET

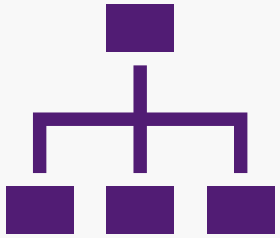
Torino Technologies Group

Xamarin

iOS & Android with C#

Native performance, native APIs, native UI

Build native mobile apps with Visual Studio



Cross-Platform

Build cross-platform UI with Xamarin.Forms

Access native APIs from shared code with Xamarin.Essentials

Share code with .NET Standard

Free & Open Source

Xamarin is part of .NET and open source on GitHub

PREVIEWS

Xamarin Developer Productivity

XAML Hot Reload

Iterate quickly on your Xamarin.Forms UIs while debugging

aka.ms/XAMLHotReload

Hot Restart

Quickly redeploy your Xamarin.Forms app without a full compile

aka.ms/HotRestart

Sessions

Xamarin

.NET Conf 2019 Keynote
https://youtu.be/1xQE2bWkwjo?list=PLReL099Y5nRd04p81Q7p5TtyjCrj9tz1t&t=1975

Xamarin.Forms: More Productive and Beautiful Than Ever
https://www.youtube.com/watch?v=iU_MUcM7BVQ&list=PLReL099Y5nRd04p81Q7p5TtyjCrj9tz1t&index=8

Building Modern Android Applications with Xamarin
https://www.youtube.com/watch?v=Za1GEmcmnI8&list=PLReL099Y5nRd04p81Q7p5TtyjCrj9tz1t&index=35

Mixed Reality with Xamarin and Azure Spatial Anchors
https://www.youtube.com/watch?v=-hJecJUZUI0&list=PLReL099Y5nRd04p81Q7p5TtyjCrj9tz1t&index=75

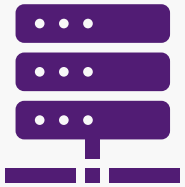
Torino Technologies Group

Web apps with Blazor

.NET

Torino Technologies Group

ASP.NET Core 3.0 Blazor



Full stack web development with C#

You don’t need to know AngularJS, React, Vue, etc.

Take advantage of stability and consistency of .NET

Runs in all browsers

Strongly typed on the client and server

Share C# code with the client and server

Web Assembly (In Preview, Release in May 2020)

Native performance

Requires no plugin or code transpilation

Torino Technologies Group

Web apps with .NET Core 3.0

Build client-side web UI with .NET

Reusable web UI components with C# and Razor

Share .NET code with client and server

Call into JavaScript libraries & browser APIs as needed

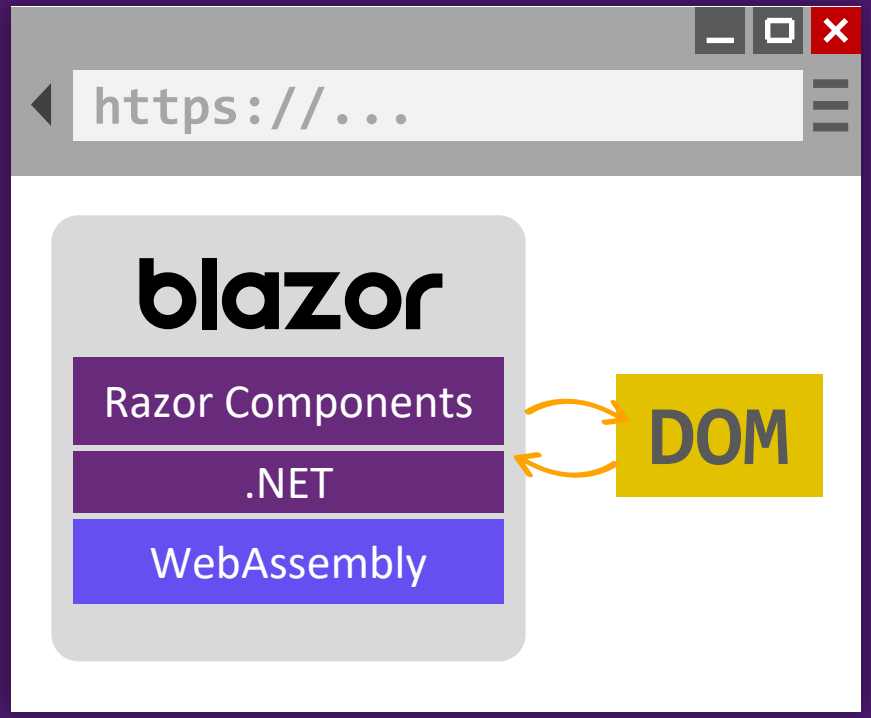
Handle client UI interactions on the server over SignalR *or* directly in the browser via WebAssembly (May 2020)

https://blazor.net

Torino Technologies Group

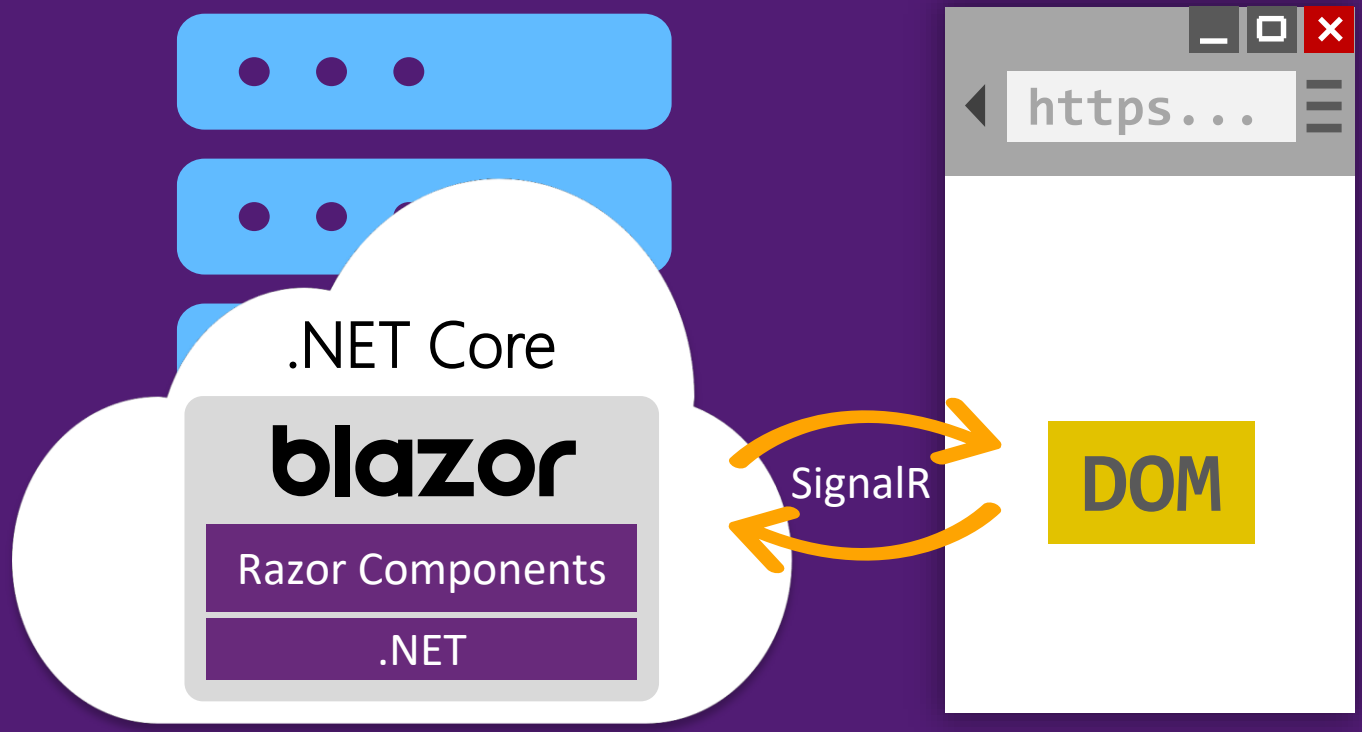
Blazor on client or server

Blazor WebAssembly



May 2020

Blazor Server



.NET Core 3.0

Torino Technologies Group

Blazor on client or server

Blazor WebAssembly

Pro:
- True SPA, full interactivity
- Utilize client resources
- Supports offline, static sites, PWA scenarios

Con:
- Larger download size
- Requires WebAssembly
- Still in preview

May 2020

Blazor Server

Pro:
- Smaller download size, faster load time
- Running on fully featured .NET runtime
- Code never leaves the server
- Simplified architecture

Con:
- Latency
- No offline support
- Consumes more server resources

.NET Core 3.0

Get started with Blazor

- Go to https://blazor.net
- Install .NET Core 3.0
- Install the Blazor WebAssembly template
- (Windows) Install Visual Studio 2019 16.3
- (Mac/Linux) Install Visual Studio Code with the C# extension

Blazor roadmap

- Blazor Server – Shipped with .NET Core 3.0
- Blazor WebAssembly – May 2020
- Blazor PWA & Electron – Previews with .NET 5
- Blazor Native – Under investigation

Blazor Server

Web app
Every interaction handled on server
Prerendered HTML (optional)

Blazor WebAssembly

Web app with client-side execution
Loaded from web server
Can work offline via Service Worker

Blazor PWA – OS installed

Appears as native app (own window)
Works offline or online

Blazor Hybrid

Native .NET renders to Electron / WebView
Appears as native app (own window)
Works offline or online

Blazor Native

Same programming model, but
rendering non-HTML UI

Torino Technologies Group

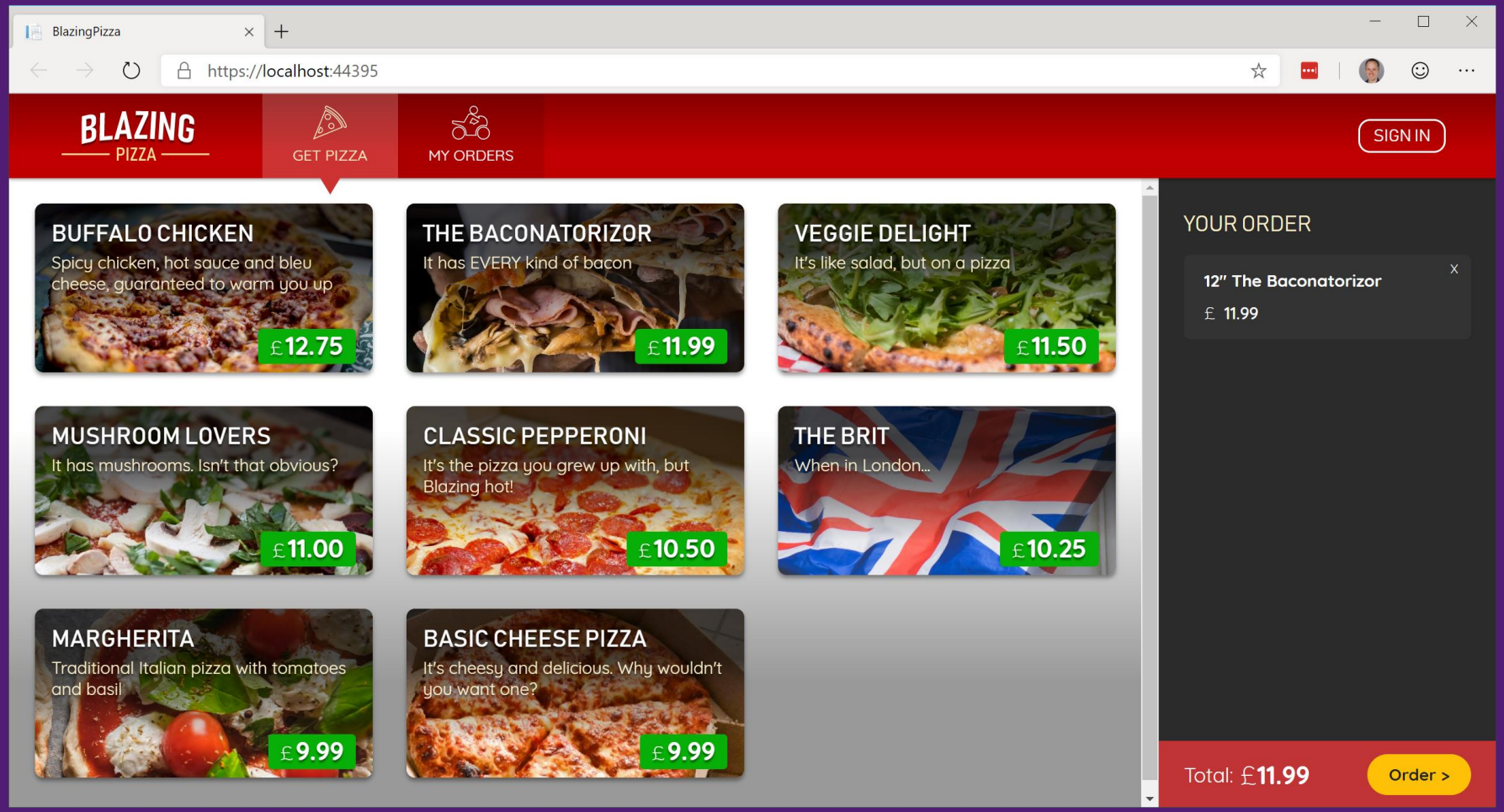
The "Awesome Blazor" community

- https://aka.ms/awesomeblazor
- Free open-source components & JS interop libraries
- Lots of fun sample Blazor apps
- Articles, videos, blogs, and other learning materials
- Chat with the Blazor community on Gitter: https://gitter.im/aspnet/blazor

DEMO

Blazor

Build your own pizza store UI with Blazor



https://aka.ms/blazorworkshop

Torino Technologies Group

Blasteroids

https://aka.ms/blasteroids

https://github.com/aesalazar/AsteroidsWasm

Torino Technologies Group

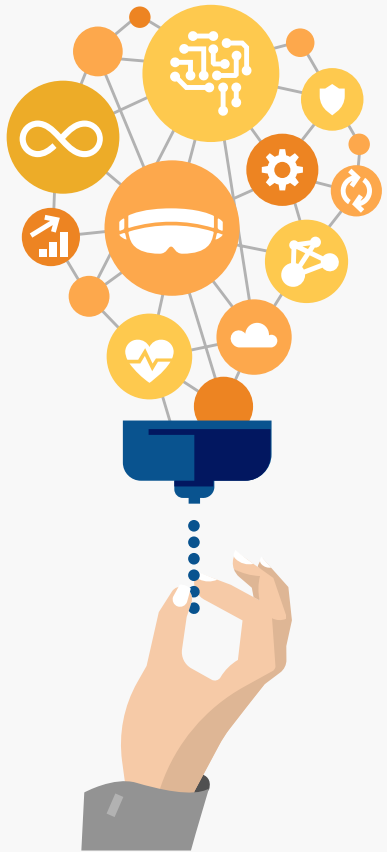
Machine Learning

.NET

Torino Technologies Group

ML.NET

An open source and cross-platform machine learning framework



Built for .NET developers

Custom ML made easy with tools

Extended with TensorFlow & more

Trusted & proven at scale

http://dot.net/ml

Torino Technologies Group

ML.NET runs anywhere

Supported Frameworks:

- .NET Core *(Natively)*
- .NET Framework *(Natively)*
- Python with *NimbusML* *(Python bindings)*

Supported processor arch.

- x64
- x86

Torino Technologies Group

ML.NET

Built for .NET developers

Create custom ML models using C# or F# without having to leave the .NET ecosystem

Custom ML made easy with AutoML

Visual Studio Model Builder and CLI make it super easy to build custom ML Models

Extended with TensorFlow & more

Leverage other popular ML frameworks (TensorFlow, ONNX, and more)

A few things you can do with ML.NET …

Sentiment analysis
Analyze the sentiment of customer reviews using a binary classification algorithm.

Sentiment analysis sample >

Product recommendation
Recommend products based on purchase history using a matrix factorization algorithm.

Product recommendation sample >

Price prediction
Predict taxi fares based on distance traveled etc. using a regression algorithm.

Price prediction sample >

Customer segmentation
Identify groups of customers with similar profiles using a clustering algorithm.

Customer segmentation sample >

GitHub labeler
Suggest the GitHub label for new issues using a multi-class classification algorithm.

GitHub labeler sample >

Fraud detection
Detect fraudulent credit card transactions using a binary classification algorithm.

Fraud detection sample >

Spam detection
Flag text messages as spam using a binary classification algorithm.

Spam detection sample >

Image classification
Classify images (e.g. broccoli vs pizza) using a TensorFlow deep learning algorithm.

Image classification sample >

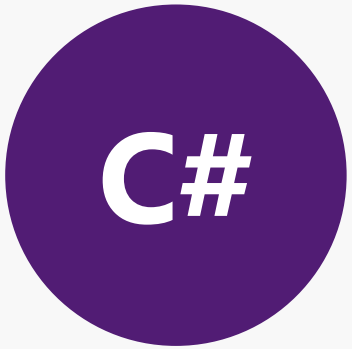
Sales forecasting
Forecast future sales for products using a regression algorithm.

Sales forecasting sample >

And more! Samples @ https://github.com/dotnet/machinelearning-samples

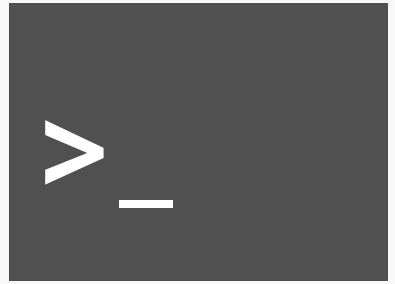
Torino Technologies Group

Three ways to use ML.NET…



ML.NET
API
(Code)

ML.NET
Model Builder
(Visual Studio UI)

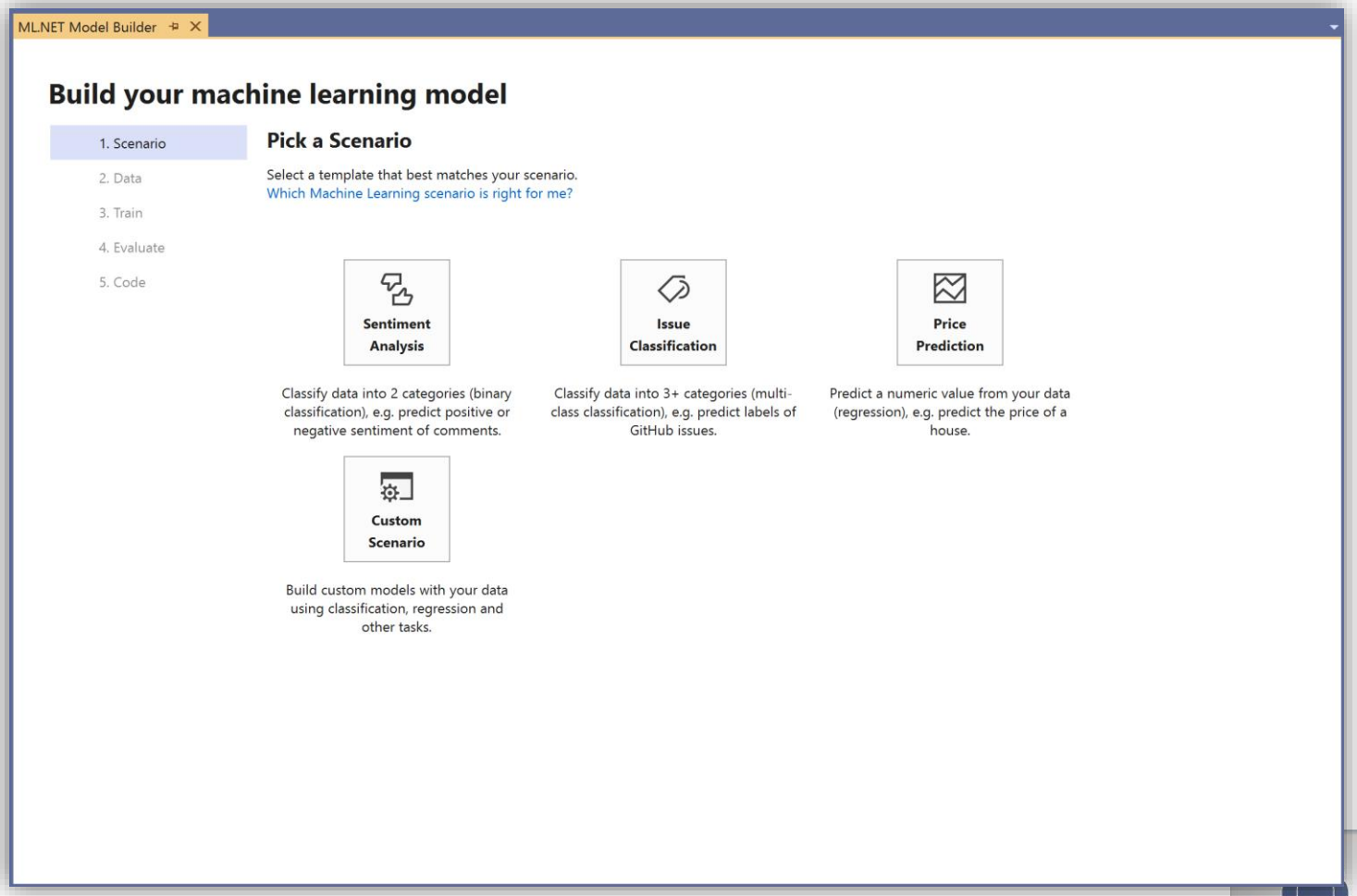


ML.NET
CLI
(Command-Line Interface)

ML.NET Model Builder
Approachable machine learning in Visual Studio

- A simple UI to easily build custom ML models with Automated ML
- Load from files and databases
- Generate code for training and consumption
- Run everything local

Download VS VSIX:
http://aka.ms/mlnetmodelbuilder

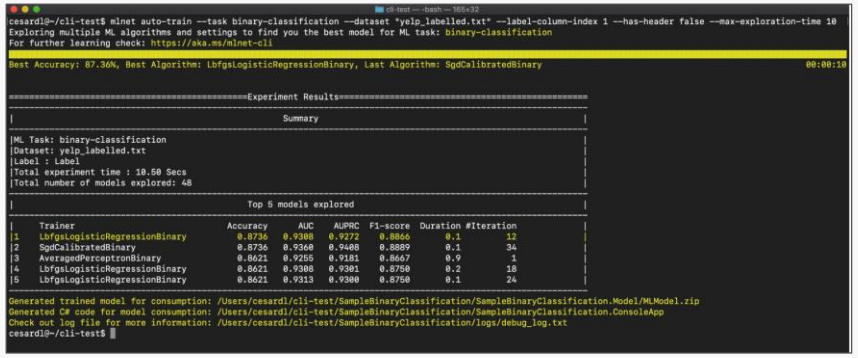


Torino Technologies Group

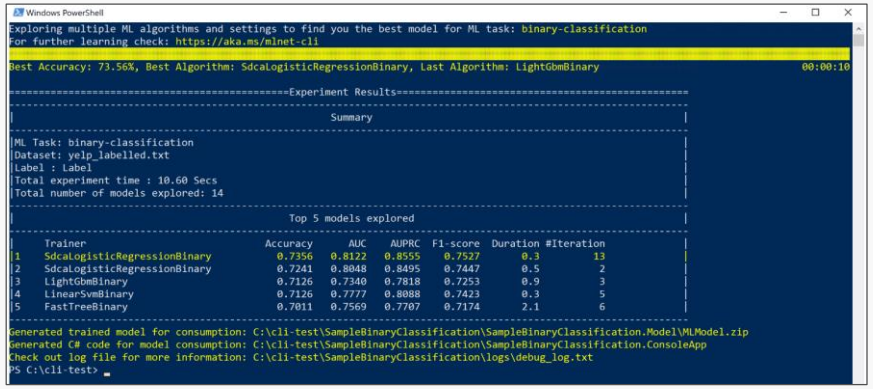
ML.NET CLI

- Use the CLI to easily build custom ML models with Automated ML
- Cross platform (Windows, Linux, MacOS)
- Generate code for training & consumption

macOS / Linux (Bash)



Windows (PowerShell and CMD)




```
> dotnet tool install -g mlnet
```

```
> mlnet auto-train --ml-task binary-classification --dataset "customer-reviews.tsv" --label-column-name Sentiment
```

Torino Technologies Group

What's new in ML.NET since May 2019?

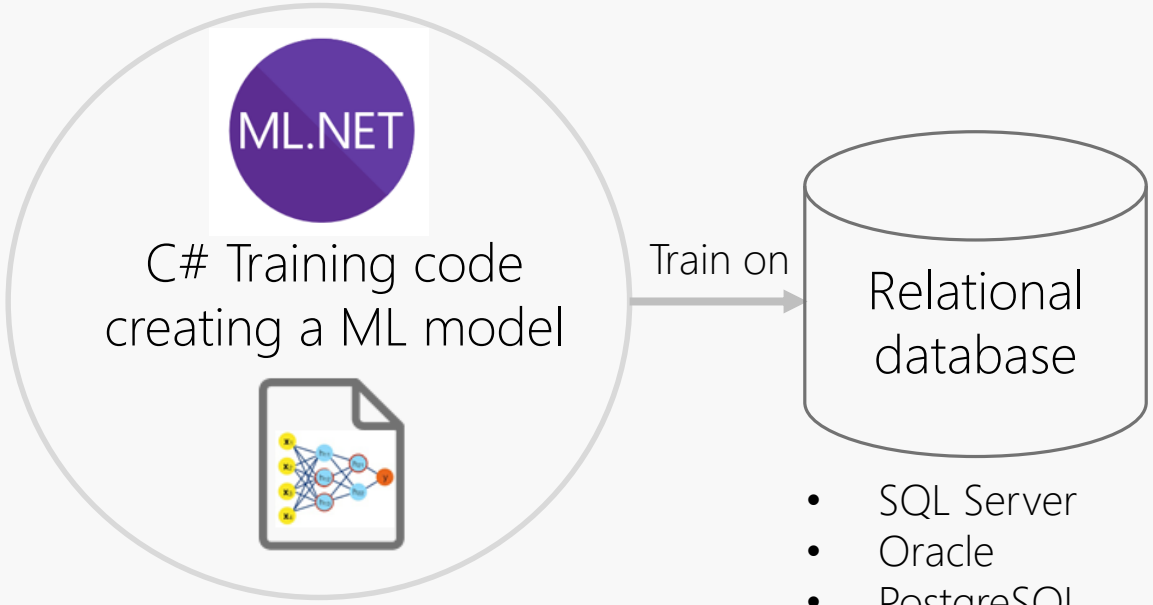
.NET

Torino Technologies Group

Database Loader

Enabled scenarios

- Training directly against relational databases.
- Simple and out-of-the-box code
- Supports any RDBMS supported by System.Data
- Currently in preview (1.4-preview release)



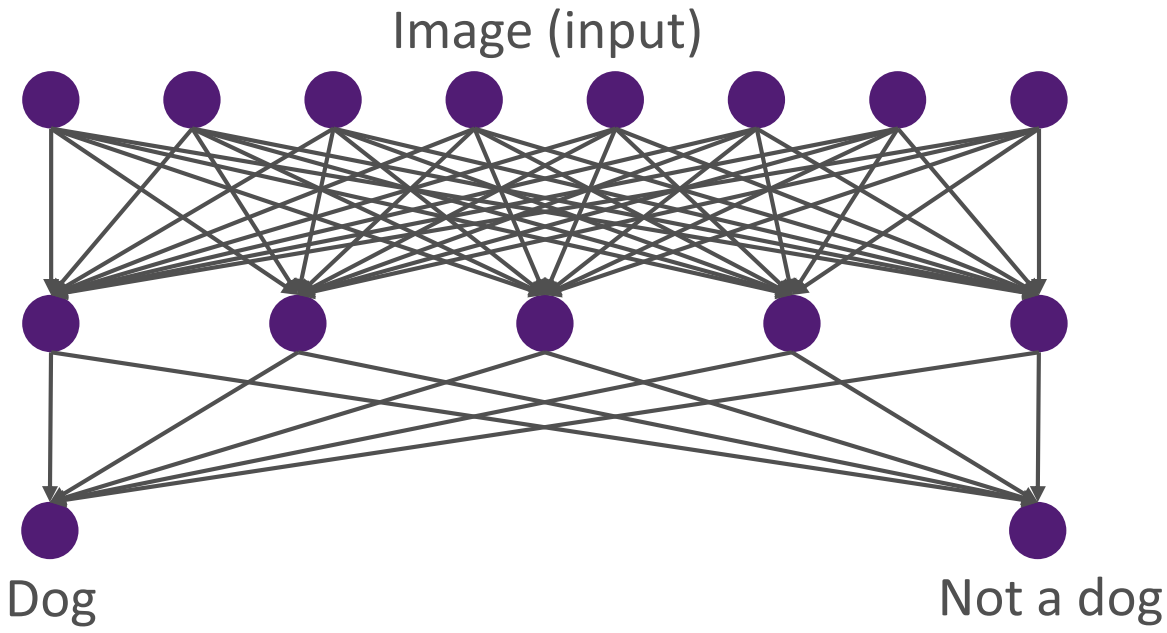
Torino Technologies Group

Use Deep Learning models with ML.NET

- Add intelligence based on DNN-based models to your .NET apps
- Enables computer vision and many more Deep Learning domains

Dog

Not a dog



Torino Technologies Group

Leading libraries in deep learning and interoperability

Microsoft's strategy is to integrate those low level libraries and runtimes into ML.NET:

Currently supported by ML.NET:
- TensorFlow
- ONNX

In ML.NET long-term roadmap:
- Torch

Leading deep learning ‘pre-trained models’
(DNN architectures)

Computer Vision

Image classification

- Google InceptionV3
- Microsoft ResNet
- NASNet
- Oxford VGG Model
- MobileNetV2
- etc.

Object detection

- Yolo (You Only Look Once)
- R-CNN
- SPP-net
- Fast R-CNN
- Faster R-CNN
- etc.

Audio and speech

- Wavenet
- espnet
- waveblow
- deepspeech2
- loop
- tacotron
- etc.

NLP (Natural language processing)

Generative Models

… other domains …

- Huge investment/cost in DNN architecture research plus costly training on large datasets made by organizations such as Google, Microsoft, Facebook, Universities and researchers.
- You can take advantage of it by simply consuming pre-trained models

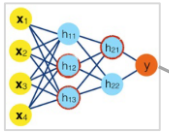
https://modelzoo.co/categories

Torino Technologies Group

Consuming pre-trained deep learning models with ML.NET

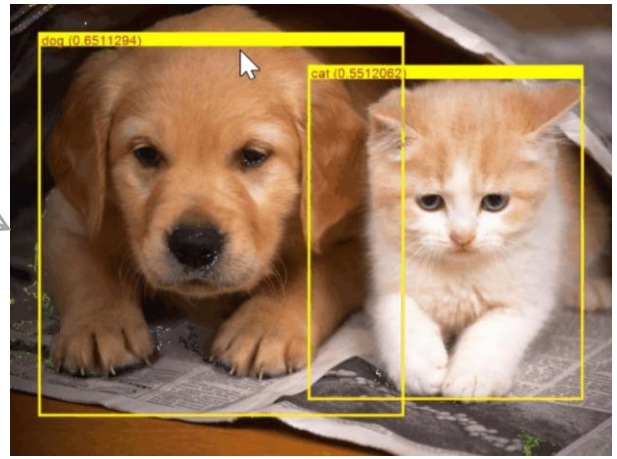
Scenario: **Object Detection** (Consuming the model)

Pre-trained
Deep learning model
(i.e. Object Detection)



Examples of pre-trained models for obj. detection:

- **YOLO** (You Only Look Once)
- **Faster R-CNN**
- **SSD**, etc.



[GitHub sample here]

Torino Technologies Group

Other important features in ML.NET

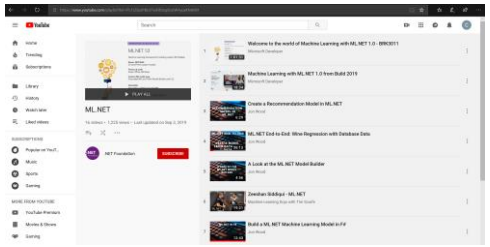
- **DevOps & ML Model lifecycle with Azure DevOps CI/CD pipelines**
- **Model Explainability and Feature input variables importance**
- **Cross-validation of ML.NET models:**
- **Loading data from 'anywhere' through the LoadFromEnumerable() method**
- **Using huge and sparse datasets (thousands or millions of columns)**
- **Deploy ML.NET model into high scalable and multi-threaded ASP.NET Core apps and services**
- **Deploy ML.NET model into an Azure Function**

- Plus many more ML Tasks and scenarios (see samples here: https://github.com/dotnet/machinelearning-samples) such as:

  Sentiment analysis, Spam detection, Fraud Detection, text classification, products recommendations, data spike detection, clustering, ranking results, etc.

Check out many of those scenarios at the **ML.NET YouTube playlist**:

https://aka.ms/mlnetyoutube



Torino Technologies Group

DEMO

ML.NET

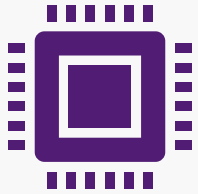
Torino Technologies Group

IoT

.NET

Torino Technologies Group

.NET Core 3.0 IoT Support



Supports Raspberry Pi and other devices

You can now run .NET Core apps in small places, including ASP.NET

Read sensor data & write to displays

New APIs for GPIO pins that enable using millions of IoT peripherals

Works with containers

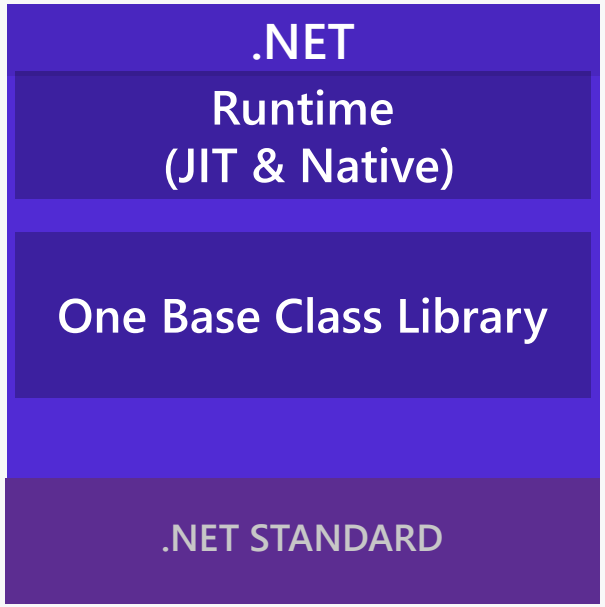
Deploy apps directly onto devices or with containers

.NET 5

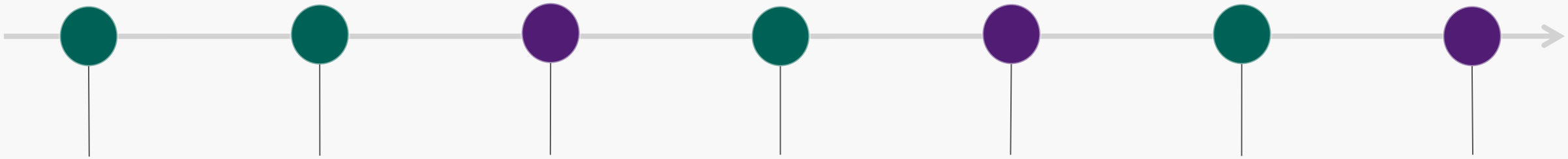
.NET

Torino Technologies Group

.NET 5



.NET Schedule



July 2019
.NET Core 3.0
RC

Sept 2019
.NET Core 3.0

Nov 2019
.NET Core 3.1
LTS

Nov 2020
.NET 5.0

Nov 2021
.NET 6.0
LTS

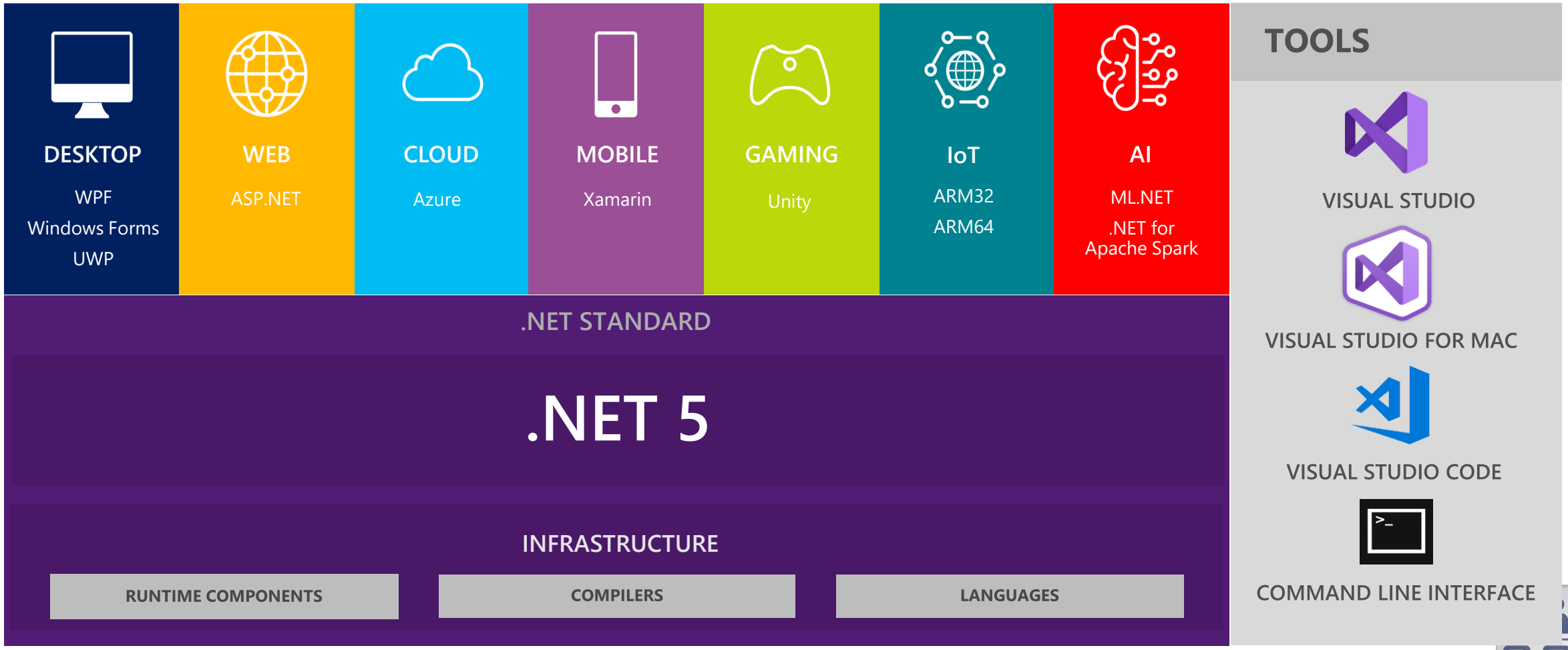
Nov 2022
.NET 7.0

Nov 2023
.NET 8.0
LTS

- .NET Core 3.0 released today!
- .NET Core 3.1 = Long Term Support (LTS)
- .NET 5.0 release in November 2020
- Major releases every year, LTS for even numbered releases
- Predictable schedule, minor releases if needed

Torino Technologies Group

.NET – A unified platform



.NET 5

Will continue to exist:

- Open source and community-oriented on GitHub
- Cross-platform implementation
- Support for leveraging platform-specific capabilities, such as Windows Forms and WPF on Windows and the native bindings to each native platform from Xamarin.
- High performance
- Side-by-side installation
- Small project files (SDK-style)
- Capable command-line interface (CLI)
- Visual Studio, Visual Studio for Mac and Visual Studio Code integration.

.NET 5

Will be new:

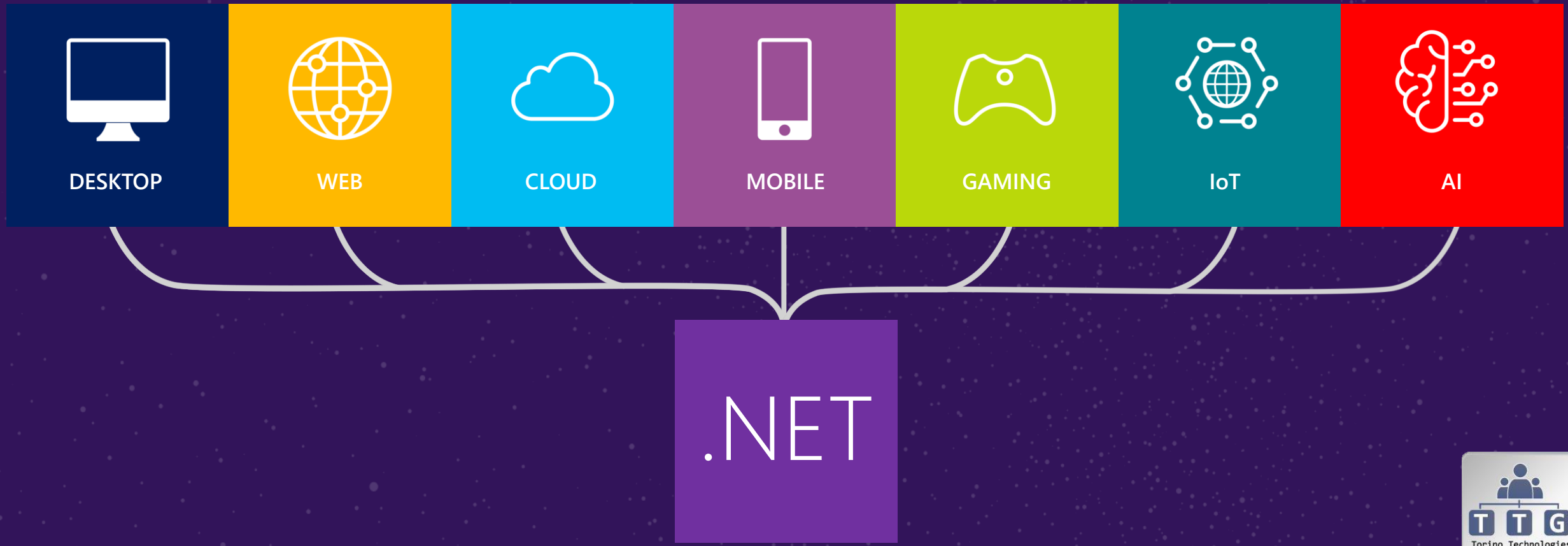
- More choice on runtime experiences (Mono, CoreCLR)
- Java interoperability will be available on all platforms
- Objective-C and Swift interoperability will be supported on multiple operating systems
- CoreFX will be extended to support static compilation of .NET (ahead-of-time – AOT), smaller footprints and support for more operating systems.

https://devblogs.microsoft.com/dotnet/introducing-net-5/

Download .NET Core 3.0 Today!

dot.net/get-core3

visualstudio.com/downloads



Thank You

ευχαριστώ Salamat Po متشكرم شكراً Grazie

благодаря ありがとうございます Kiitos Teşekkürler 谢谢

ขอบคุณครับ Obrigado شكريه Terima Kasih Dziękuję

Hvala Köszönöm Tak Dank u wel дякую Tack

Mulţumesc спасибо Danke Cám ơn Gracias

多謝晒 Ďakujem תודה நன்றி Děkuji 감사합니다

Q&A

Microsoft

Torino Technologies Group

About

Ing. Gianni ROSA GALLINA

@giannirg

R&D Specialist, Senior Software Engineer @ Deltatre

- AI, Machine Learning (Google/Microsoft ML APIs, PyTorch, fastai)
- Virtual Reality (Oculus Rift, Gear VR, WinMR, Unity 3D)
- Augmented/Mixed Reality (HoloLens, Magic Leap)
- Immersive video streaming and 3D graphics for sport events
- NUI Prototypes (Microsoft Kinect, Leap Motion)
- Mobile App developer (Windows / Android / Xamarin)
- Cloud solutions with Microsoft Azure (serverless, video workflow)

MVP Microsoft Most Valuable Professional

Microsoft Specialist
Programming in C#
Programming in HTML5 with JavaScript & CSS3

Microsoft CERTIFIED Solutions Developer
Windows Store Apps Using C#
Web Applications

Unity Certified Programmer

PLURALSIGHT Author

http://gianni.rosagallina.com

Torino Technologies Group

About

Clemente Giorio

@tinux80

R&D Specialist, Senior Software Engineer @ Deltatre

- Augmented/Mixed/Virtual Reality
- Artificial Intelligence, Machine Learning
- Internet of Things
- Embedded Apps
- Multimodal Tracking

dotNET{podcast}

MVP Microsoft Most Valuable Professional

intel Software INNOVATOR

[PACKT] PUBLISHING Author

FAB LAB NAPOLI

TTG Torino Technologies Group