



TRAILHEAD
TECHNOLOGY PARTNERS

.NET and C# Training

Hi, I'm J.

Jonathan "J." Tower

Principal Consultant & Partner

Trailhead Technology Partners

- 🏆 Microsoft MVP in .NET
- 🏆 Frequent conference speaker
- 📅 Organizer of Beer City Code conference



- ✉ jtower@trailheadtechnology.com
- 🌐 trailheadtechnology.com/blog
- 🐦 [jtowermi](https://twitter.com/jtowermi)

Introductions

1. Your name
2. Your role on the team
3. Your experience with programming or software development

More About Me

1. Jonathan "J." Tower
2. Partner & Principal Consultant, Trailhead Technology Partners
 - Founding Partner, Architect, Head of Sales & Marketing, Trainer
3. Developing professionally for 24 years
 - Started .NET in 2001
 - Microsoft MVP in .NET
 - Consultant most of my career
 - Started in Chicago
 - Started Trailhead about 7.5 years ago
 - Architect and run projects today

Introductions

1. Your name
2. Your role on the team
3. Your experience with programming or software development

Introductions

1. Your favorite aspect of software development
2. One interesting fact about you
3. What you are hoping to get out of the training
4. Your biggest fear about the training, if any

More About Me

1. I like the creative aspects of software
2. I've been to all 50 states, and 52 of the 63 US National Parks
3. I'm looking forward to teaching something I love to people who want to learn about it
4. Wanting to make sure its as useful as possible for everyone

Introductions

1. Your favorite aspect of software development
2. One interesting fact about you
3. What you are hoping to get out of the training
4. Your biggest fear about the training, if any

How It's Going To Work

- In-person today; online after
- Most Wednesday mornings this winter and spring
- We'll meet a total of 16 times
- Questions at ANY TIME
- Joe is available for questions between sessions
- Joe will arrange for office hours to help with assignments

How It's Going To Work

1. Review solution to previous assignment (15 min)
2. New topic instruction, examples (3.5 hr)
3. Review assignment for next time (<15 min)

NOTE: Assignments should take you 15 minutes to 2 hours per week

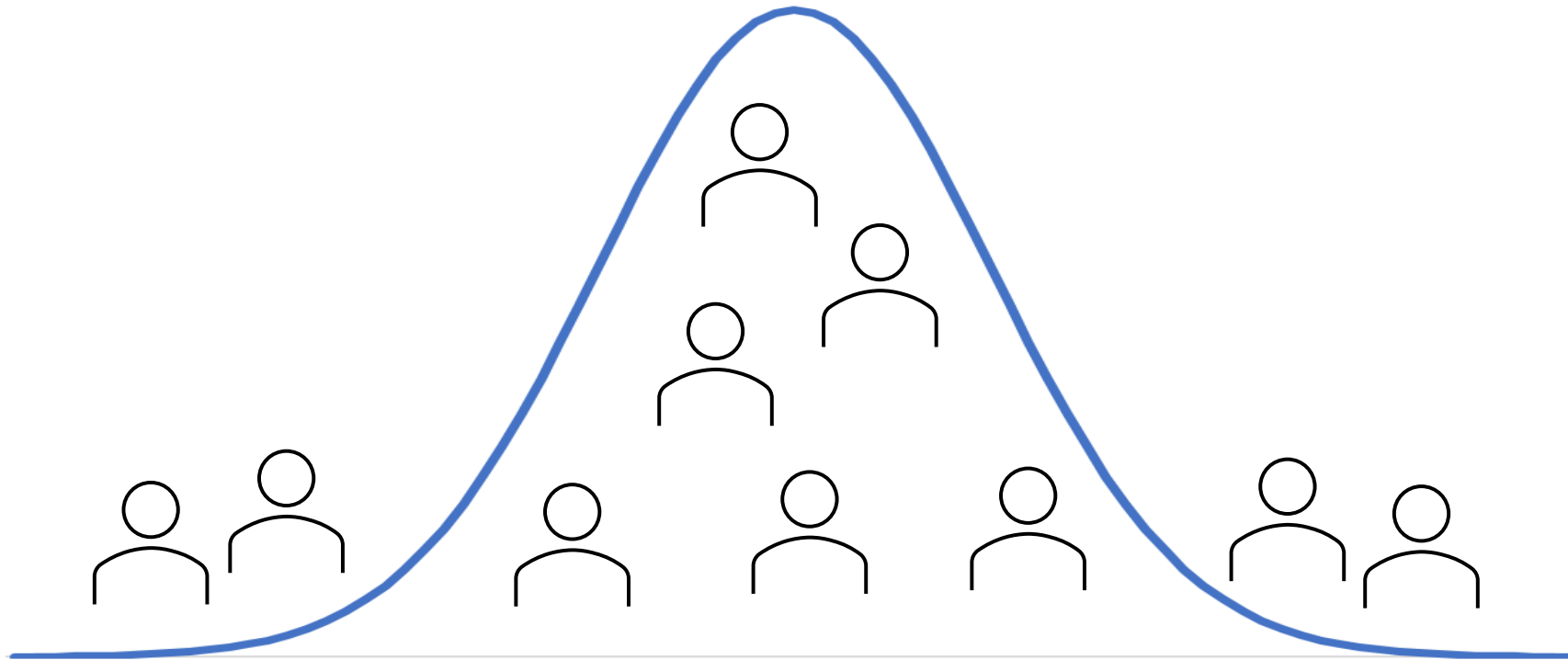
Course Agenda

1. .NET History and Overview
2. Visual Studio 2022 IDE Basics
3. Visual Studio Advanced
4. C# Introduction
5. .NET BCL Introduction
6. Data Access Basics
7. C# Advanced
8. Entity Framework
9. Version Control
10. ASP.NET Web APIs

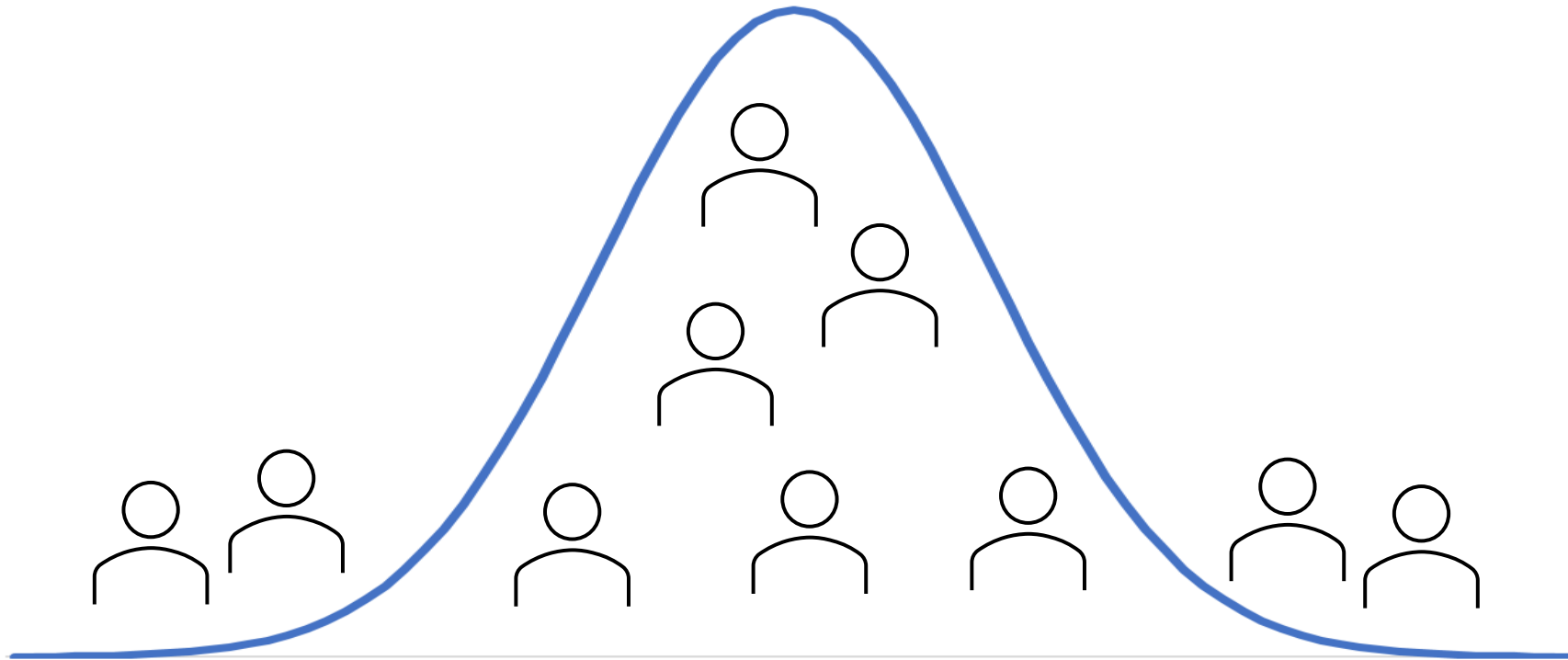
Today's Agenda

1. Format overview and expectation setting
2. .NET History and Overview
3. Visual Studio 2022 IDE Basics

Everyone Has Different Experience

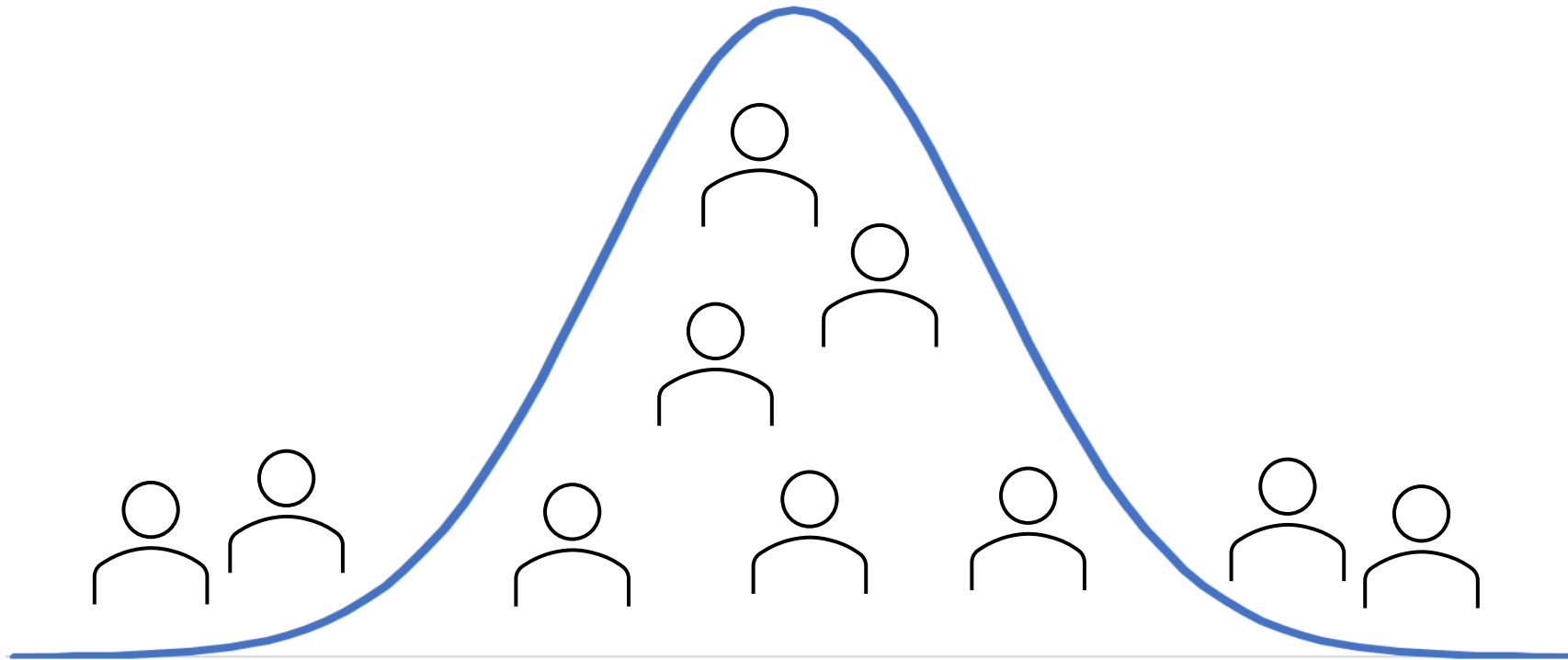


Everyone Has Different Experience



You Will Get Out Of It What You Put Into It

Everyone Has Different Experience



You Will Get Out Of It What You Put Into It

Do the assignments, turn off distractions during class, attend office hours, ask questions, fill out form after each session

.NET History and Overview

"Dot Net"

- Pronounced as "dot net"
- Written either .NET, or increasingly as dotnet
- Inspired by the modern internet in late-90's and 2000

It All Started with .NET Framework

- **.NET Framework** is a proprietary software framework developed by Microsoft that runs primarily on Microsoft Windows
- Released in 2000-2001
- Most popular languages C# and VB.NET
- Most popular IDE: Visual Studio

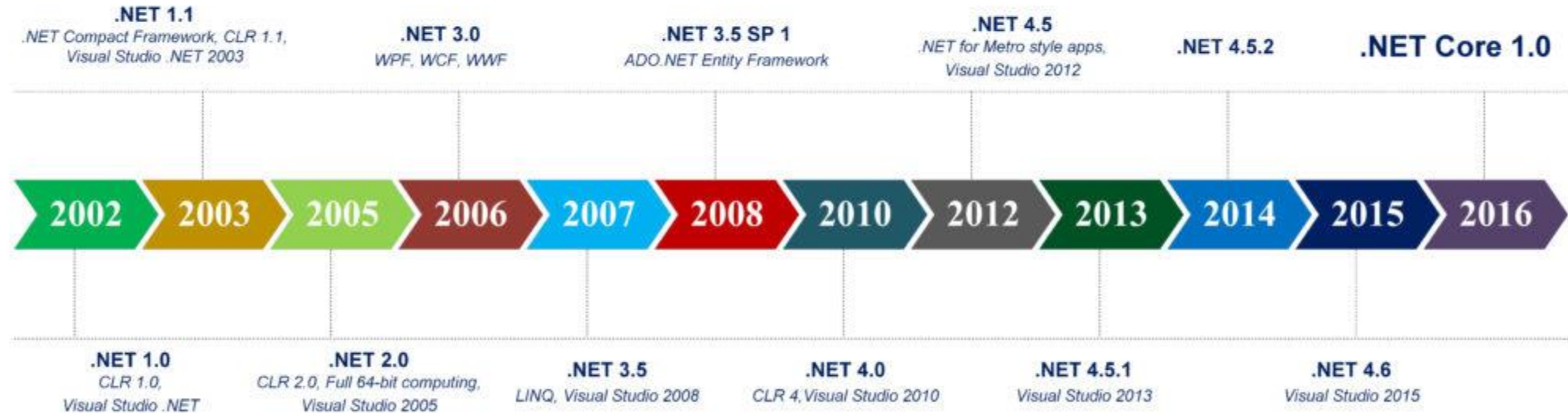
You're Coming to .NET at a Good Time

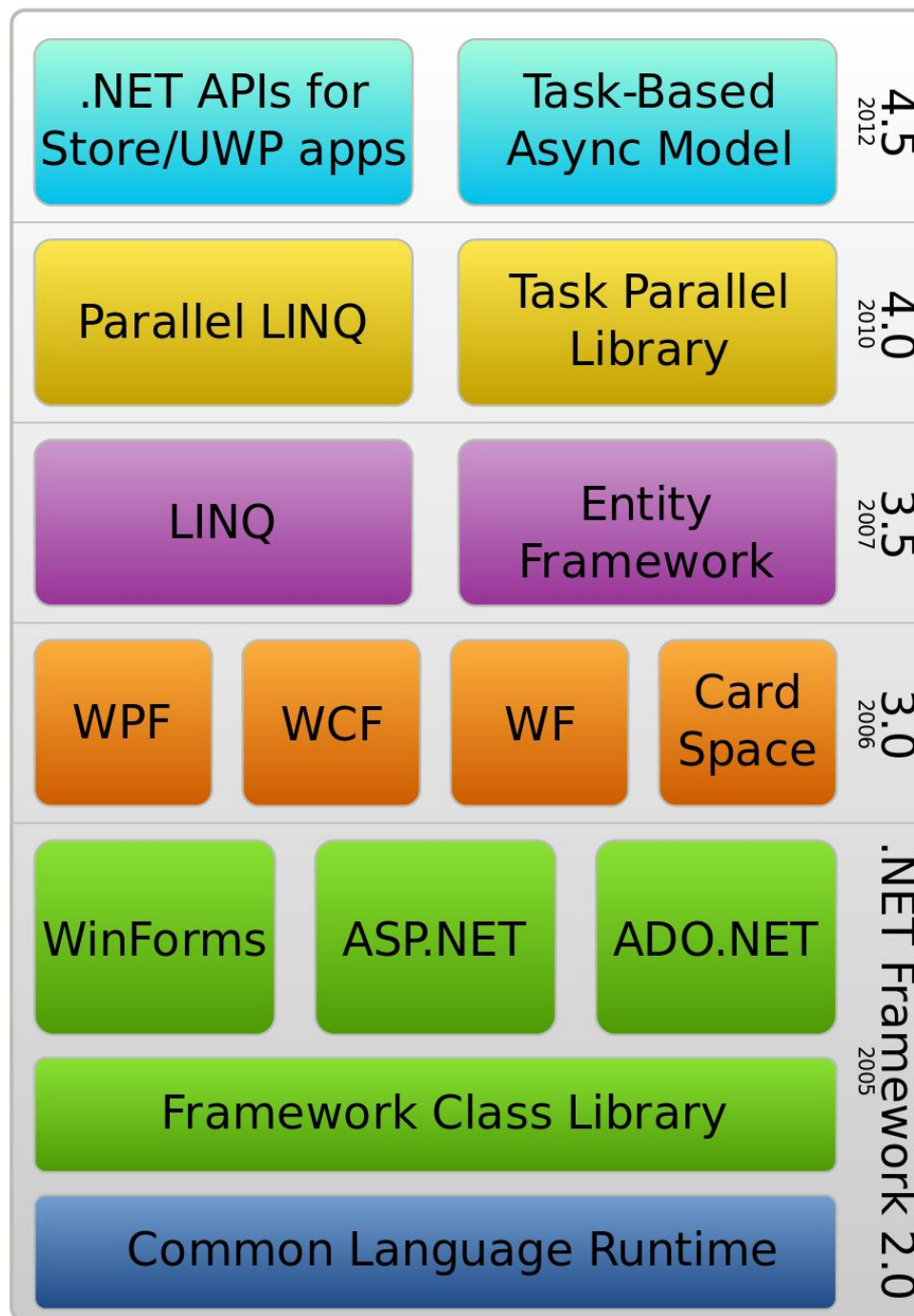
<https://dotnet.microsoft.com/en-us/platform/why-choose-dotnet>

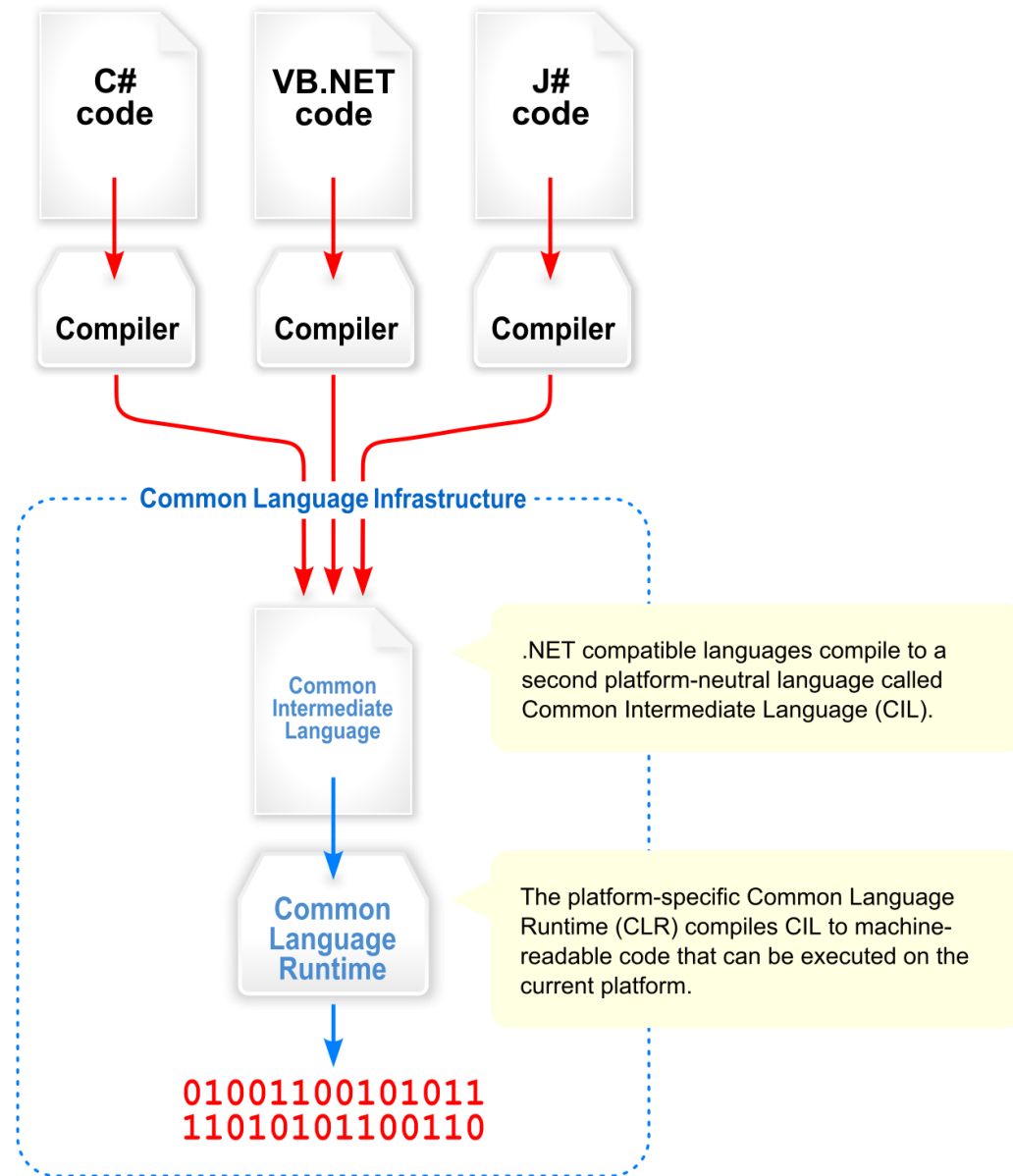
- Productive, general purpose
- Loved
- High-performance
- Secure
- Large community

NOTE: Some of these apply primarily to the newest versions of .NET

The release history of .NET Framework





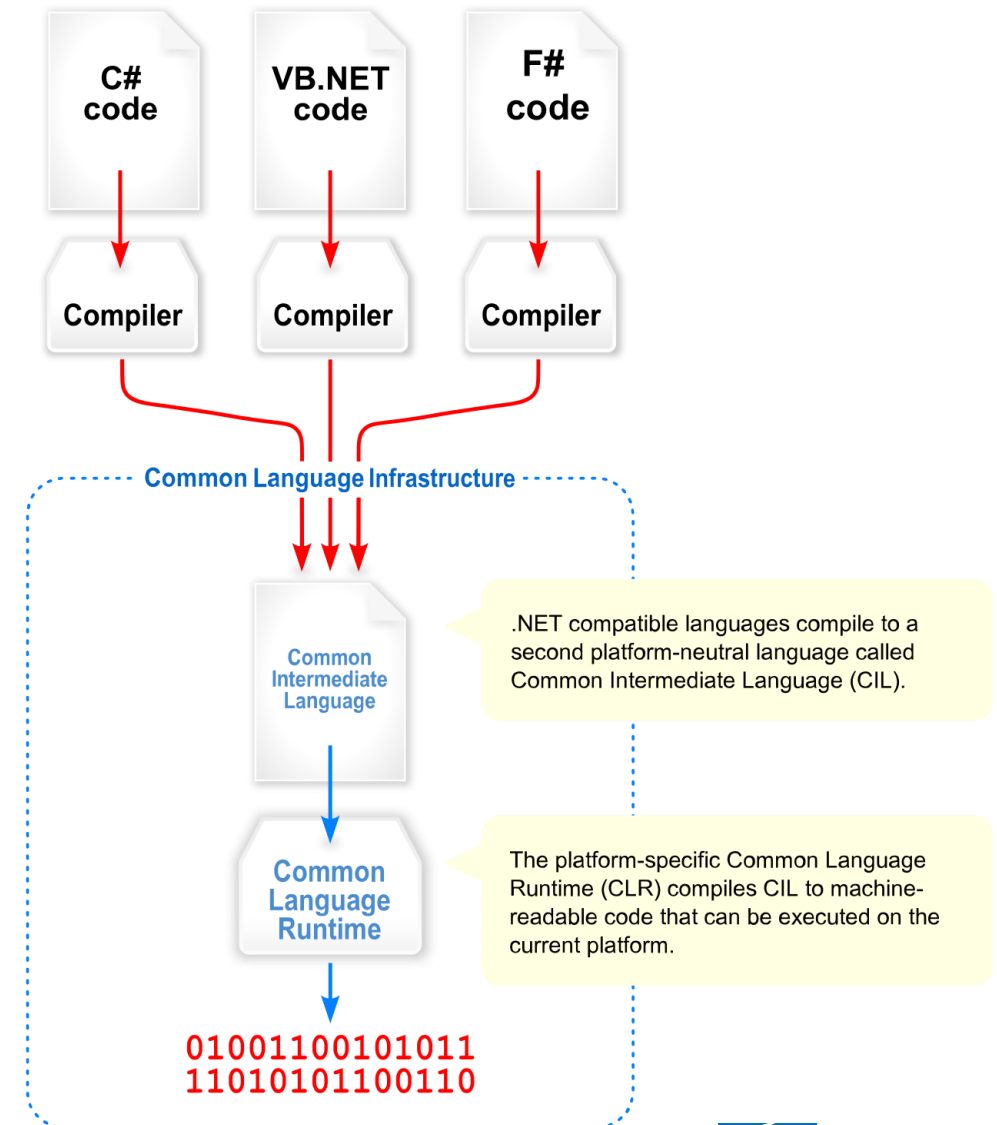


.NET Acronym Soup

- CLI and CLR
- CTS and CLS
- CSC and VBC
- CIL (MSIL)
- JIT and AoT
- BCL and FCL
- .NET Compiler Platform/Roslyn

CLR and CLI

- Common Language Runtime
- CLR is an execution engine responsible for executing and managing the lifecycle of a .NET executable
- Common Language Infrastructure
 - CTS
 - CLS
 - (And a couple of other things)



CTS and CLS

- Common Type System
- A standard that specifies how type definitions and specific values of types are represented in computer memory
- Intended to allow programs written in different languages to easily share information
- Common Language Specification is a subset of CTS
- Defines rules and restrictions a language must follow to run under .NET

CSC and VBC

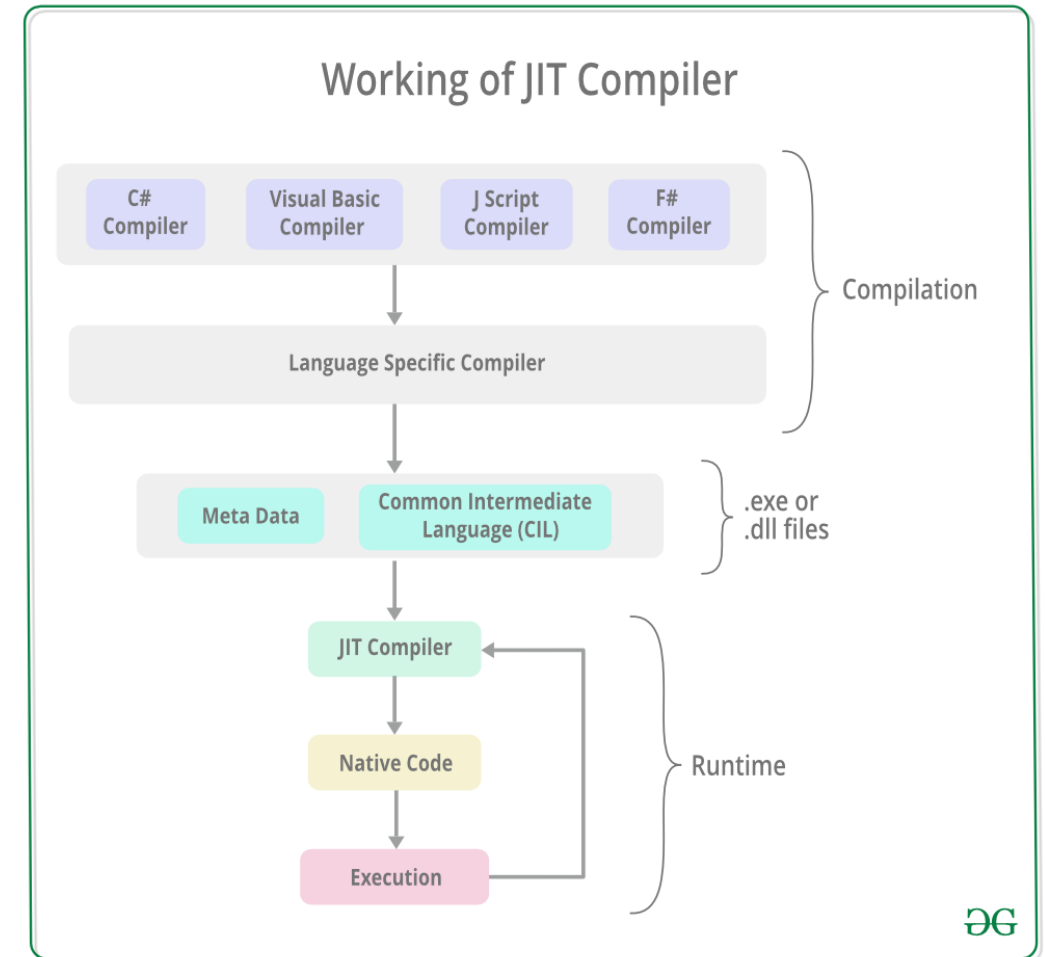
- C# compiler and VB.NET compiler
- Turn human-readable C# and VB.NET into CIL (MSIL)

CIL (Formerly MSIL)

- “Bytecode” similar to Java
- Run by the CLR
- Common Intermediate Language
- Formerly Microsoft Intermediate Language

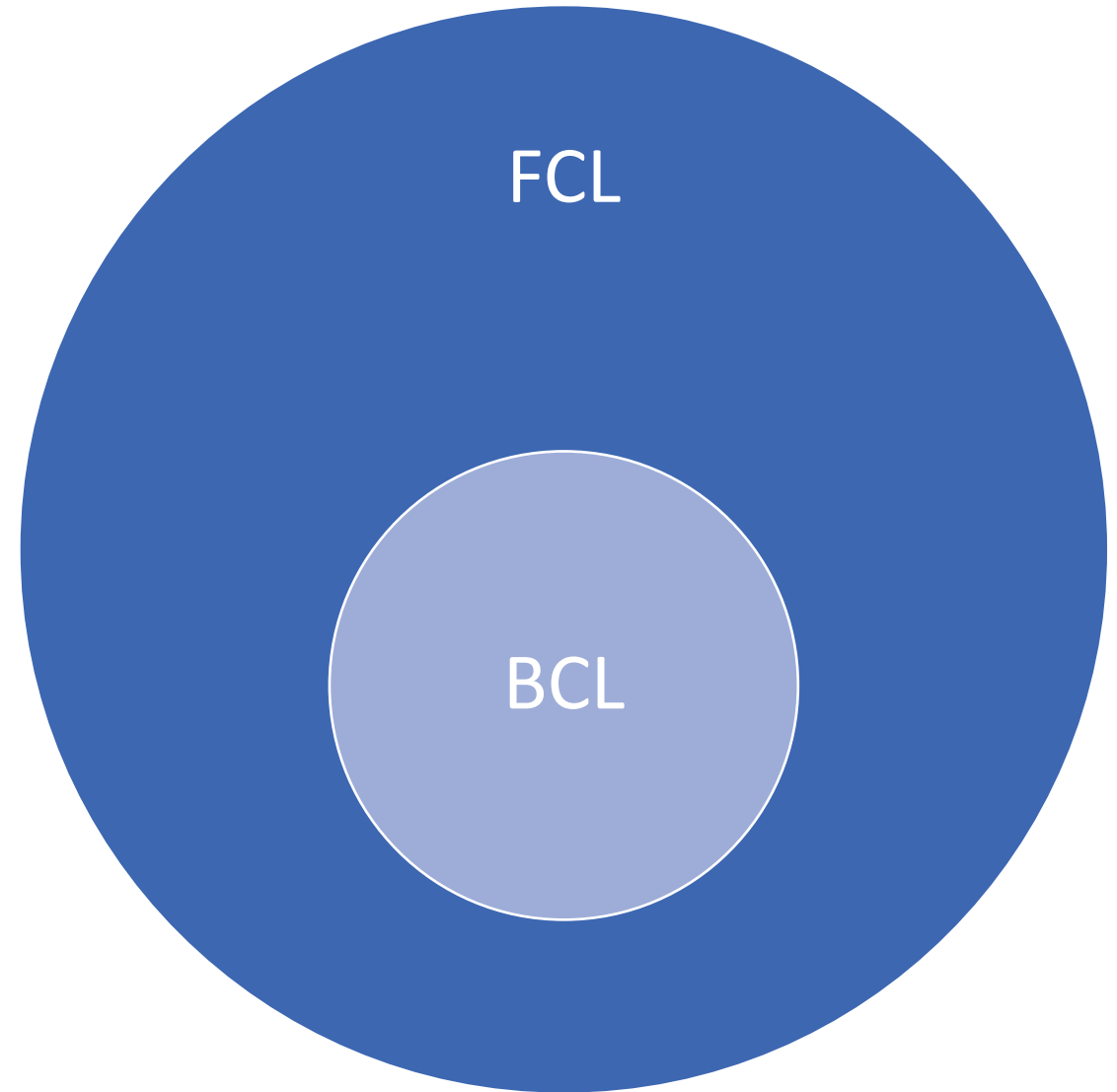
JIT and AoT

- Just-In-Time compilation
- Ahead-of-Time compilation
- JIT most common
- AoT now supported for more rare scenarios



BCL and FCL

- Base Class Library
- Framework Class Library
- BCL is a core class library
- FCL adds some advance class libraries, including ASP.NET, WinForms, the XML stack, ADO.NET and more



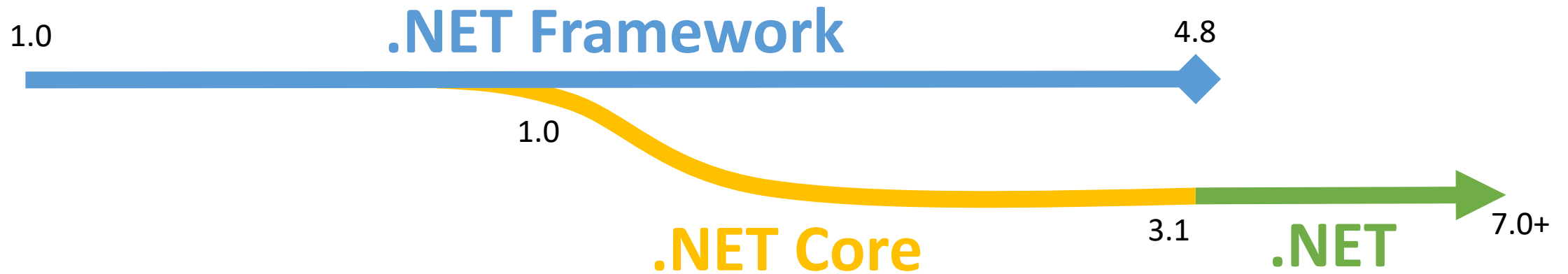
.NET Compiler Platform (Roslyn)

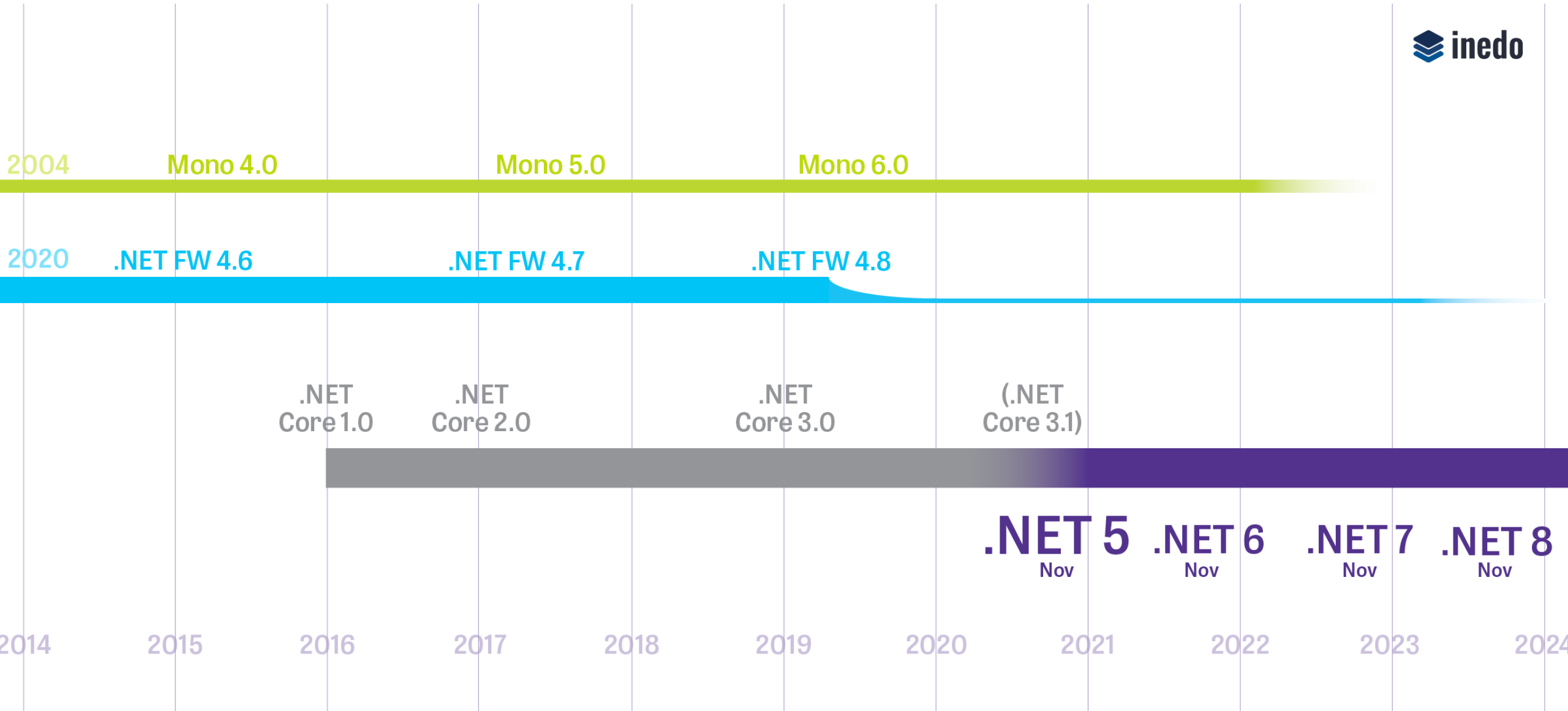
- Serves as a Compiler-as-a-Service
- Compilers for the C# and Visual Basic languages exposed as services via APIs.
- APIs for code analysis and refactoring
- Allows for
 - Compilation and checking as you type
 - Smart code generation tools
 - Powerful code analysis tools
 - Other IDEs

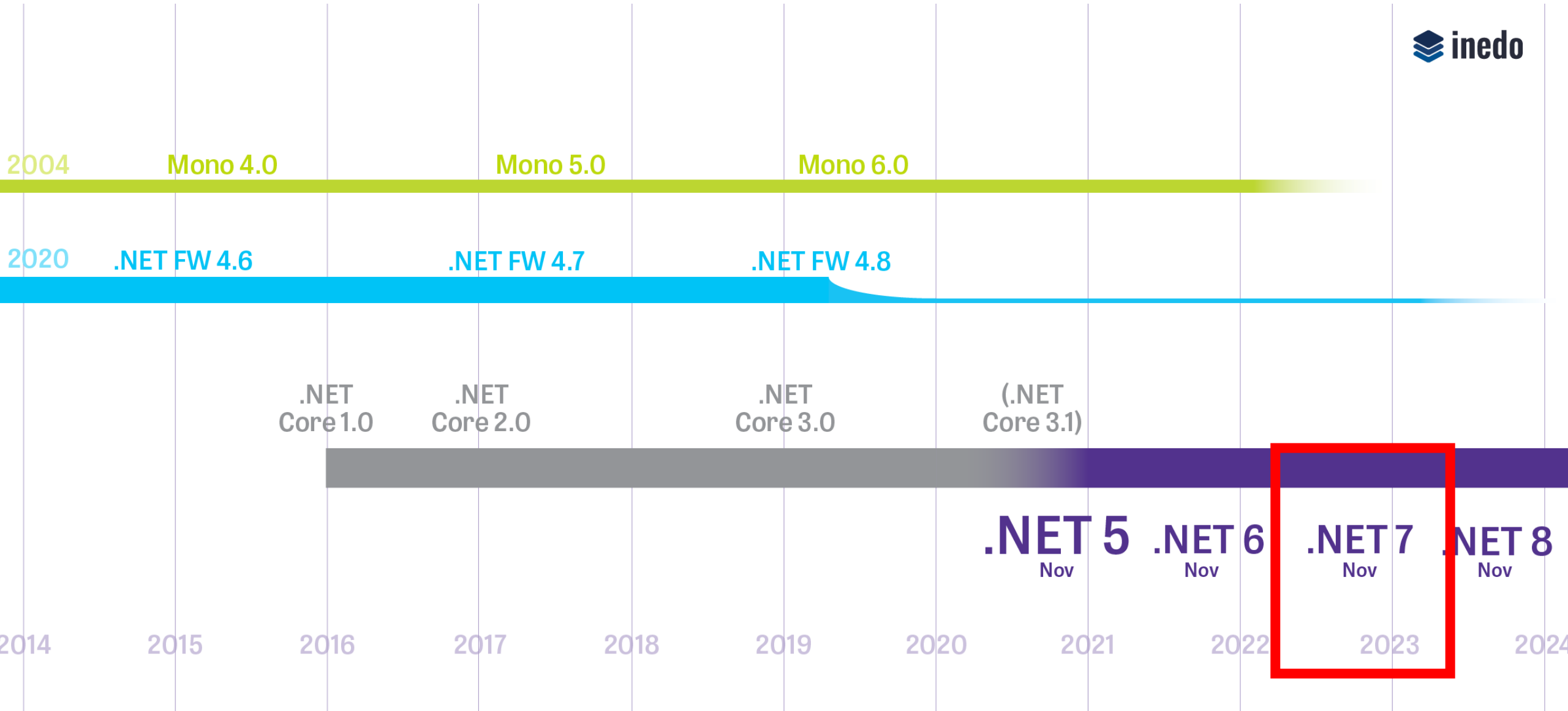
What is .NET Core

- Starting in 2016
- Completely rewritten version of .NET
- Replaces .NET Framework (which is now at EOL)
- Cross-platform: Windows, Linux, and Mac
- Performance is a feature

.NET Branding Going Forward







Tick-Tock Versioning

- Annual major version (in Nov)
- LTS (even versions) – patch and support for 3 years
- STS (odd versions) – patch and support for 18 months

Visual Studio Basics

What is an IDE?

- Integrated Development Environment
 - Projects
 - Syntax highlighting
 - Code completion
 - Refactoring
 - Debugging
 - Version control
 - Search
 - AI
- Visual Studio = Most popular IDE in the world

VS Editions

- Community, Professional, and Enterprise

Supported Features	Visual Studio Community Free download	Visual Studio Professional Buy	Visual Studio Enterprise Buy
⊕ Supported Usage Scenarios	●●●○	●●●●	●●●●
Development Platform Support ²	●●●●	●●●●	●●●●
⊕ Integrated Development Environment	●●●○	●●●○	●●●●
⊕ Advanced Debugging and Diagnostics	●●○○	●●○○	●●●●
⊕ Testing Tools	●○○○	●○○○	●●●●
⊕ Cross-platform Development	●●○○	●●○○	●●●●
⊕ Collaboration Tools and Features	●●●●	●●●●	●●●●

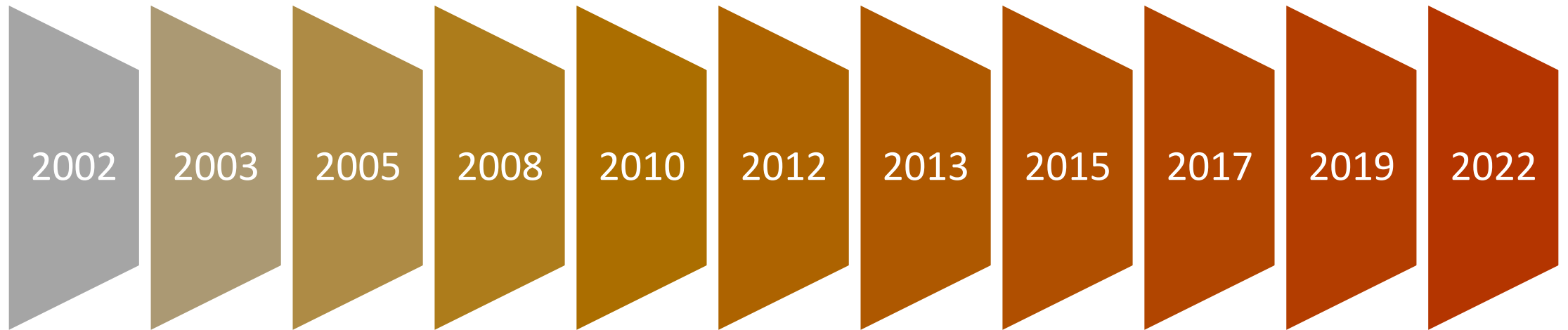
<https://visualstudio.microsoft.com/vs/compare/>

Let's Install Visual Studio Now

If You Don't Have It Already...

visualstudio.microsoft.com/downloads

Visual Studio Versions



Visual Studio vs Visual Studio Code

- Visual Studio
 - Fully featured IDE
 - Available for Windows and MacOS*
 - Paid product (excluding Community Edition)
 - **Project and solution based**
 - Compilation and debugging built in
- Visual Studio Code
 - Light-weight editor
 - Windows, Linux and MacOS
 - Free
 - **File and folder based**
 - Compilation or debugging using plug-ins

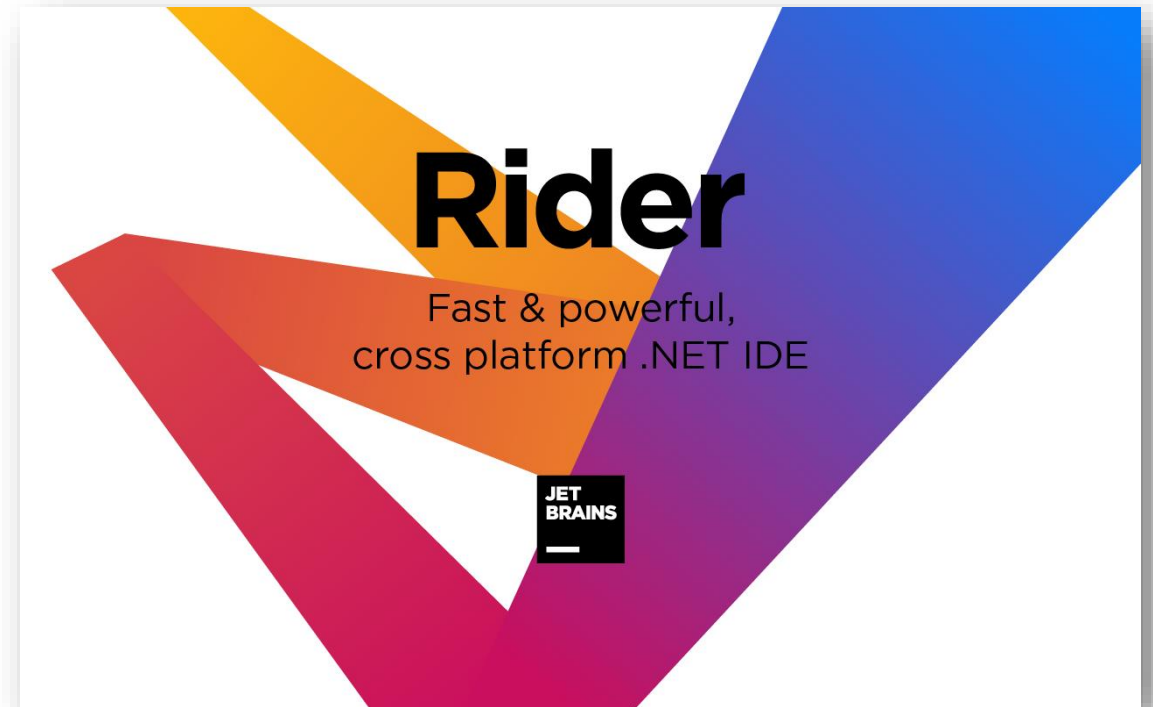
Visual Studio for Mac

- Supports .NET Core and later
- Supports some project types
- Supports Xamarin projects on Mono

JetBrain's Rider

- Visual Studio alternative
- Fast
- Paid product

<https://www.jetbrains.com/rider/>



Project and Solution Structure

- Solution file – group of projects edited/compiled together
- Project file – group of files compiled into single output
- Folders & Solutions Folders

Compilation and Assemblies

- Assemblies
 - DLL – libraries used by executables at run-time
 - EXE – directly executable
- Selected build configuration
- /bin and /obj directories

Demo

Project Structure and Compilation

ASP.NET = .NET Web Development

- Replaced older Active Server Pages (ASP) technology
- Started with "Web Forms" to make web development feel like desktop Development (Win Forms)
- Eventually replaced with other models: MVC, Web API, Razor Pages, and more
- Web Forms didn't come forward into ASP.NET Core
- Other web frameworks did come forward

Demo

ASP.NET

Desktop Development

- Console (command-line, text only)
- WinForms
- Windows Presentation Foundation, or WPF (XML-based UIs)
- Universal Windows Platform (UWP)
- WinUI
- .NET MAUI - Multi-platform App UI

Demo

WinForms

Class Library

- Shared code
- Only runs in the context of another project using it
- Can be compiled against .NET Fwk, .NET Core, .NET, or “.NET Standard”

Demo

ASP.NET + WinForms + Class Library

Many Other Project Types

- JavaScript SPAs
- Unit Testing
- UI Automation Testing
- Windows Service
- Xbox/Video Games
- Database
- Cloud-Native Technologies
- Blazor

IDE Panes

- Documents
- Solution Explorer
- Toolbox
- Properties
- Output
- Object Browser

Demos

Commonly Used IDE Panes

Assignment 01

- Create a .NET console “Hello world” application using the code below

```
using System;
namespace MyConsoleApp
{
    internal class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Hello, world");
            Console.ReadKey();
        }
    }
}
```

- Run the application in the IDE
- Build the application and run the .EXE from outside the IDE

<https://github.com/jonathantower/learning-dotnet>