

# .NET



- Sourcen mit Beispielen zum Skript finden sie unter <https://github.com/florianwachs/FHRWebservices>



## Online

<https://dot.net>

<https://docs.microsoft.com>

<https://www.microsoft.com/net/learn>

MSDN C# Programmierhandbuch (bitte docs nehmen)



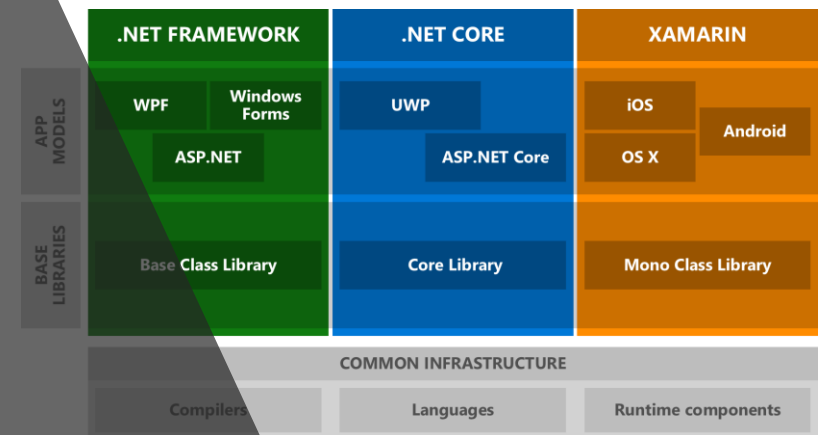
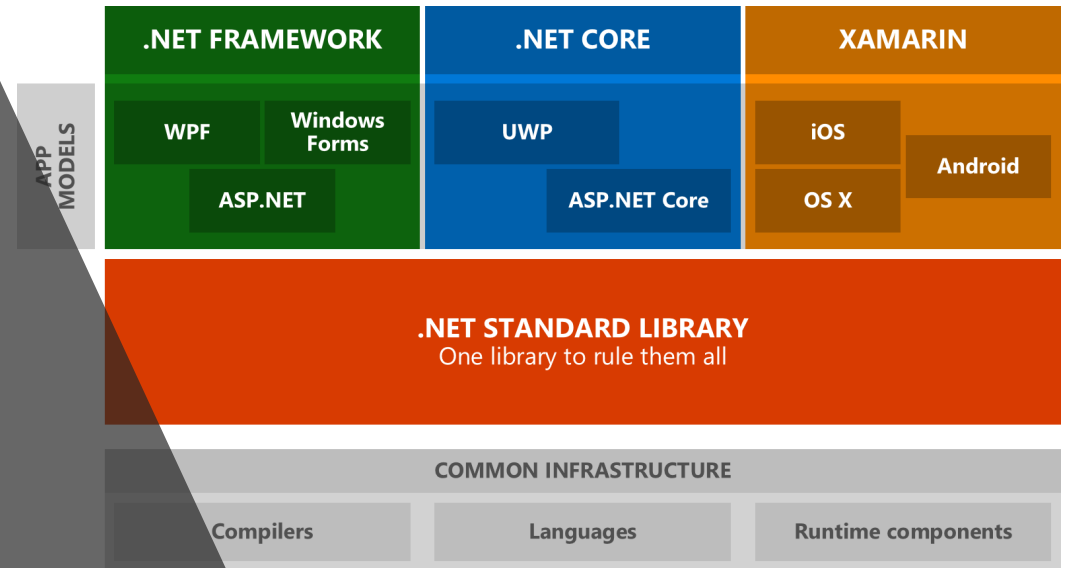
## Bücher

C# 7.0 in a Nutshell: The Definitive Reference, Joseph Albahari & Ben Albahari

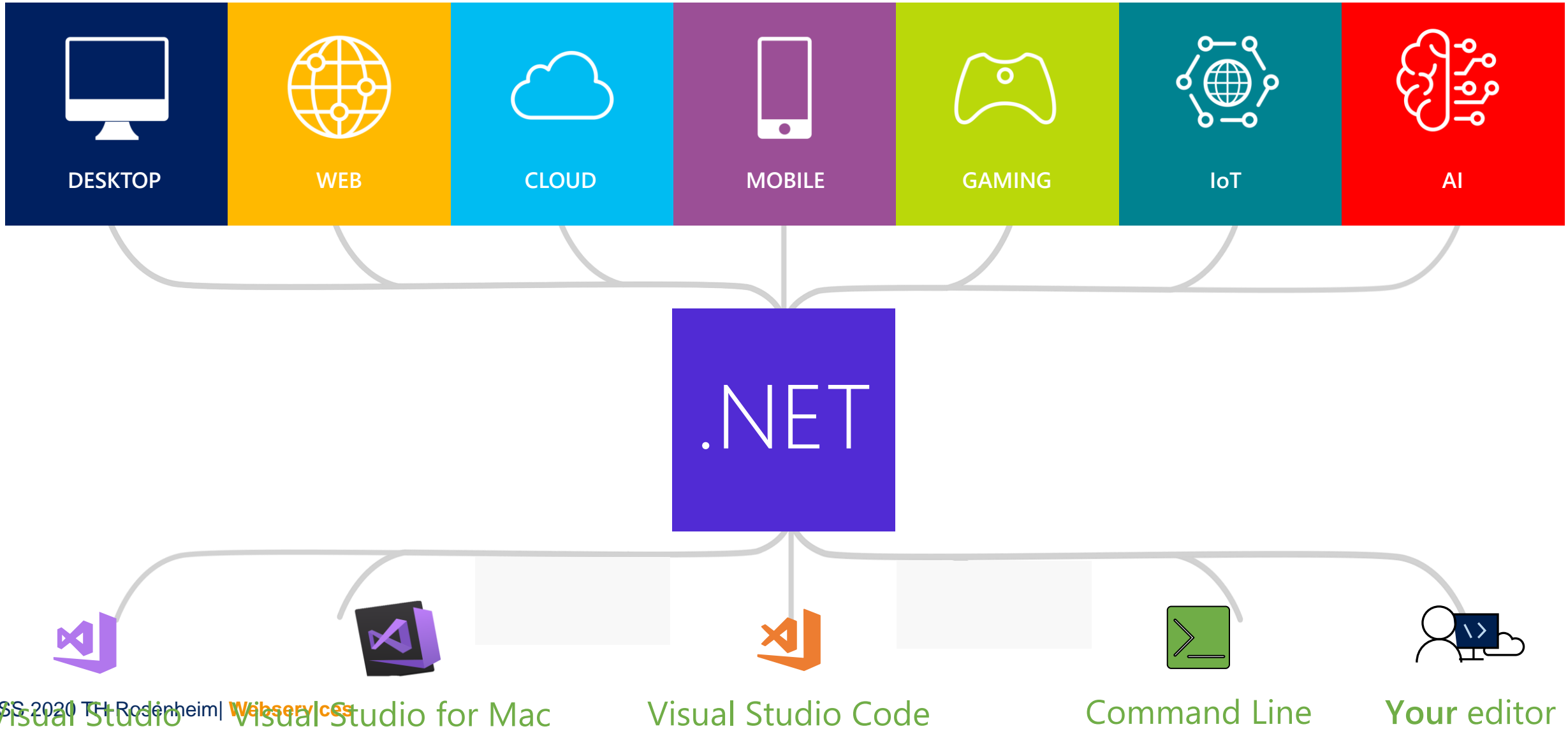
C# in Depth, Jon Skeet

Pro C# 6.0 and the .NET 4.6 Framework (Expert's Voice in .NET), Andrew Troelson

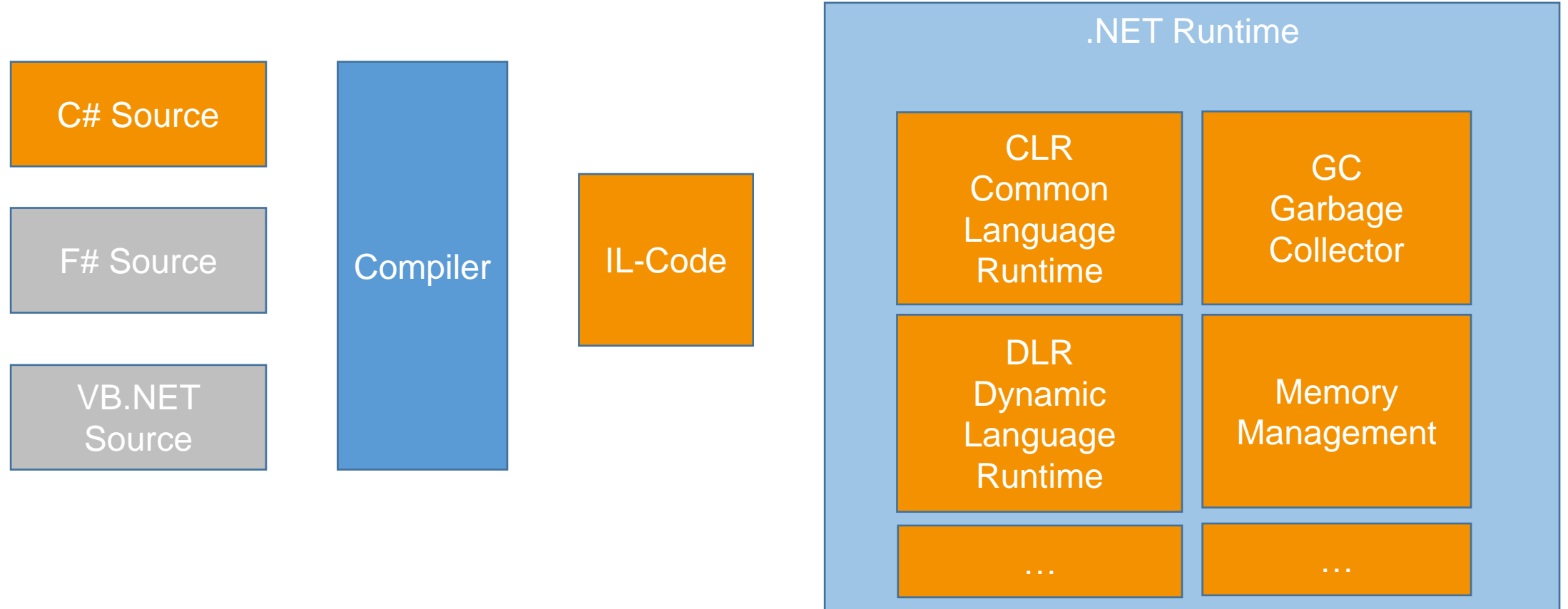
# .NET



# Your platform for building **anything**



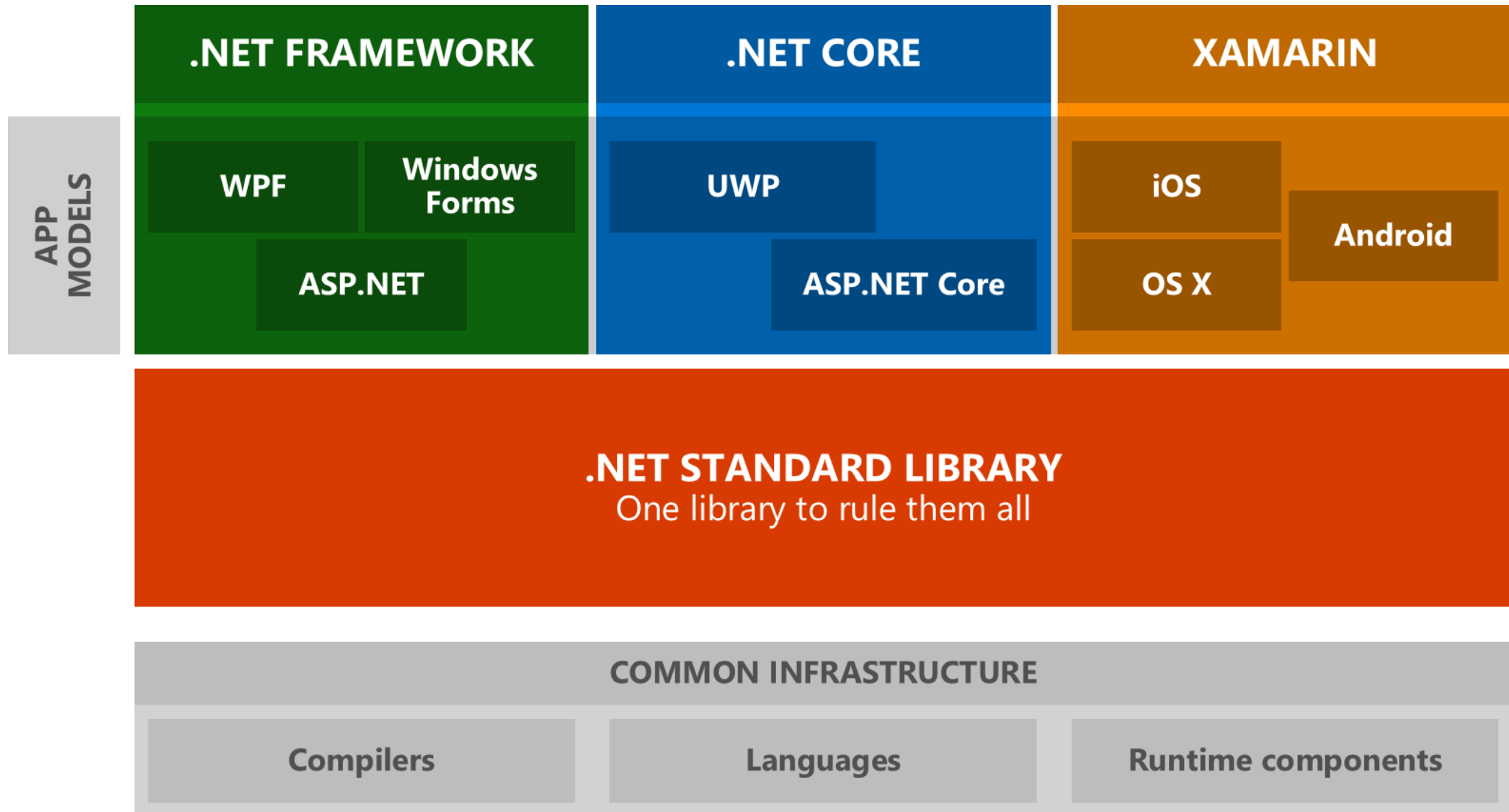
# .NET PLATFORM



- C# soll eine mächtige aber leicht(er) zu erlernende Sprache sein
- Im Vergleich zu C++ deutlich entschlackt und typische Fehlerquellen wurden entschärft (Pointer, Mehrfachvererbung, Speicherverwaltung)
- Das .NET-Framework soll für ein breites Spektrum an Aufgabenstellungen bereits passende Lösungen bieten (Base Class Library)
- Plattformunabhängigkeit
- Sprachneutralität (Managed C++, F#, IronPython)

- Hohe Developer-Produktivität durch effektive Tools
- Die .NET-Runtime ist für's Debuggen ausgelegt und optimiert
- Aktive Weiterentwicklung der Sprache
- Performance (RyuJIT, Multithreaded Generational GC, ...)
- Verbesserung am Deployment
- Ständige Anpassung an neue Herausforderungen wie Cloud und Internet of Things (IoT)



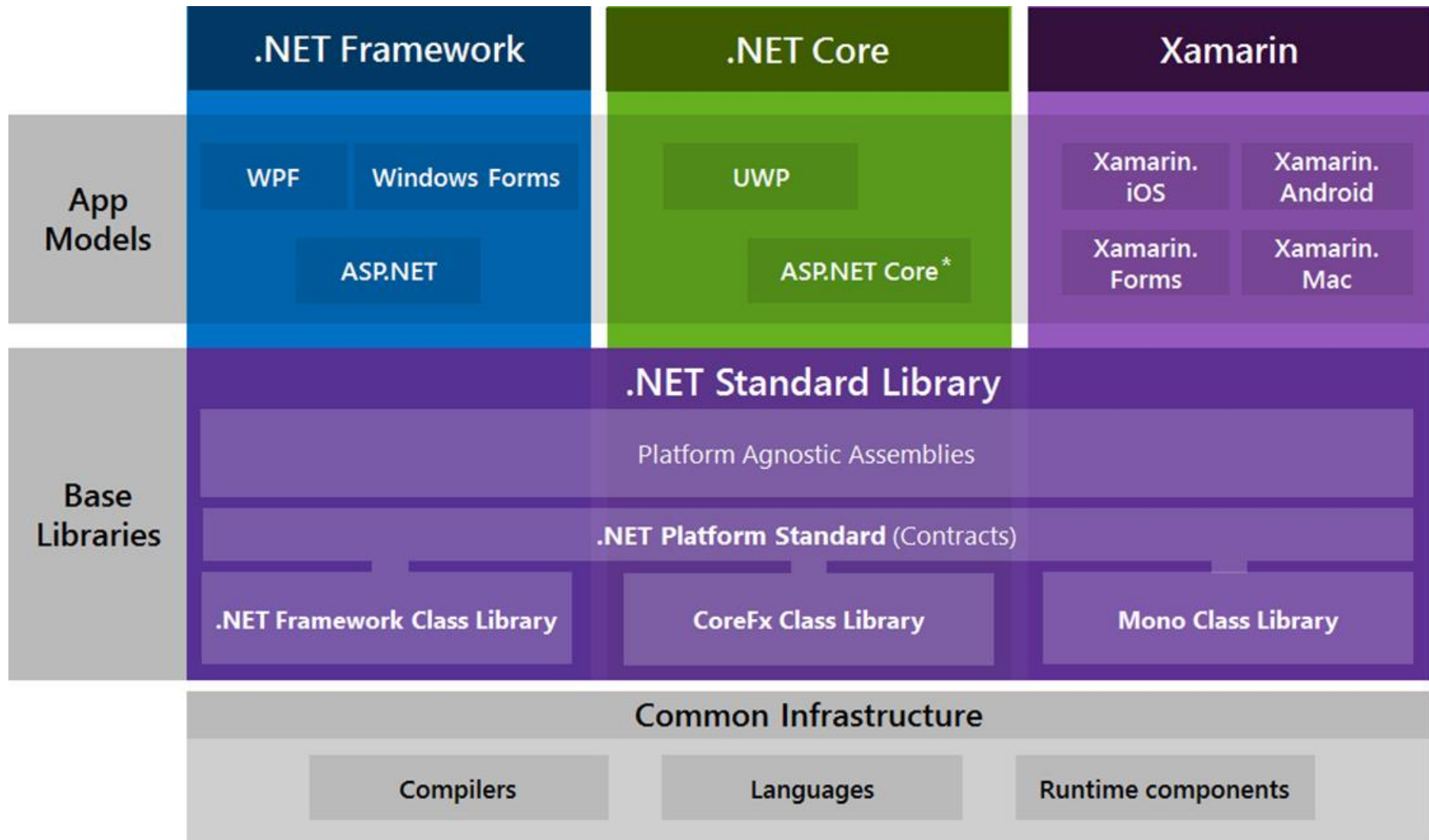


# .NET STANDARD

.NET Platform	.NET Standard			
	1.0	1.1	1.2	1.3
.NET Core	→	→	→	→
.NET Framework	→	4.5	4.5.1	4.6
Xamarin.iOS	→	→	→	→
Xamarin.Android	→	→	→	→
Universal Windows Platform	→	→	→	→
Windows	→	8.0	8.1	
Windows Phone	→	→	8.1	
Windows Phone Silverlight	8.0			

- Ziel ist eine einheitliche API-Oberfläche welche von den verschiedenen API-Runtimes unterstützt wird
- Es wird empfohlen den aktuellen Standard 2.0 zu verwenden

# FULL FRAMEWORK VS .NET CORE



## .NET – A unified platform



# .NET Core 2.2 Major Features

## .NET Core

- Multi-tier JIT compilation (opt-in)
- SQL Connection token auth
- Windows ARM32 support (Nanoserver and IoT Core)

## EF Core

- Spatial extensions for SQL Server and SQLite providers
- Reverse engineering of database views
- Collections of owned entities
- Query tagging

## ASP.NET Core

- Template updates: Bootstrap 4, Angular 6
- Web API improvements, including API security
- HTTP/2
- IIS in-process hosting
- Health checks
- Endpoint routing
- SignalR Java client now available (.NET Client in 2.1)

# .NET Core 3.1 LTS Major Features

## .NET Core

- Multi-tier JIT compilation (default)
- C# 8
- GPIO Support Raspberry Pi
- ARM64 Linux Support
- Hardware Intrinsics

## EF Core

- Many Bugfixes

## ASP.NET Core

- Endpoint Routing
- High Performance JSON Serialization (früher JSON.NET)
- Blazor WebAssembly Preview
- Performance

# FULL FRAMEWORK VS .NET CORE

## .NET Framework (4.8)

- Seit 2002 öffentlich verfügbar
- Vereint mehrere Entwicklungsplattformen wie ASP.NET, WCF, WPF, Workflow, Winforms, ADO.NET
- Hat direkte Abhängigkeiten zu Windows
- Sourcecode öffentlich einsehbar aber nicht Open Source (Code Open Prinzip)

## .NET Core 2.2 (3.0 in Preview)

- Kompletter Rewrite
- Cross-Platform (Windows, Linux, Mac) von Anfang an, weitere Ports (Raspberry, Tizen)
- Alle Teile (Compiler, Library, Frameworks) sind Open Source und werden auf Github bereitgestellt

# .NET FULL FRAMEWORK EINSATZGEBIETE (2020)

- bestehende .NET Anwendungen
- 3rd Party Libraries die (noch) nicht für .Net Core verfügbar sind
- Technologien die nicht auf .NET Core verfügbar sind (WebForms, WCF, WPF\*, WinForms\*)

\* Verfügbar mit .NET Core 3.0+



# .NET CORE EINSATZGEBIETE (2020)

- Cross-Platform (Windows, Linux, Mac und mehr)
- Microservice Architekturen
- (Docker) Container
- High Performance
- Side-by-Side Versionierung auf Applikationslevel
- Neuentwicklungen werden hier vorangetrieben

# .NET CORE EINSATZGEBIETE (2020)

- Ab .net Core 3.0 auch WinForms und WPF Anwendungen (nur auf Windows)
- Self-Contained Deployment (.net core 2.0+)
- Single-Exe-Deployment (.net core 3.0+)

- 2020 kommt .net (core) 5\*
- Full Framework geht in „Legacy-Mode“
- Neue Versionen von C# setzen neueste .NET Version voraus
- Performanter Interop mit Swift / Objective-C und Java
- AOT Compiler von Mono
- Eine Base Class Library und .NET Runtime

- Tutorials & Blogs: Immer auf .net core 2.0+ achten
- <https://docs.microsoft.com> ist eine sehr gute Informationsquelle
- .NET ist open source und für verschiedenste Workloads geeignet

- Visual Studio 2019
- Visual Studio for Mac
- Visual Studio Code (mit C# Extension)
- JetBrains Rider
- Vim, ...