

What's New in ASP.NET Core 6.0

Get Set up to Work with .NET 6.0

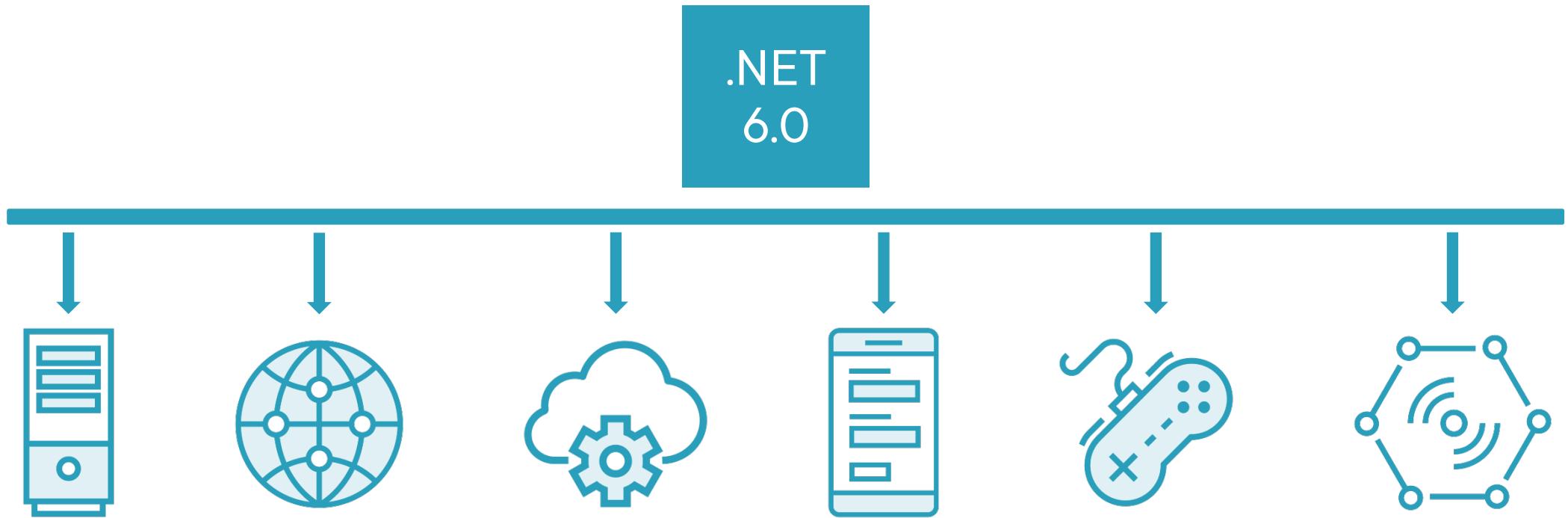


Alex Wolf

www.thecodewolf.com



The Power of the .NET Platform





A Quick Note

.NET 5.0 and .NET 6.0 share many goals and themes.



.NET 6.0 achieves the vision of a unified .NET ecosystem.



Common Questions in the .NET Ecosystem



Eager .NET Developer



Should I use ASP.NET Framework or ASP.NET Core?



How does the .NET Standard versioning system work?




What is Entity Framework vs Entity Framework Core?




A Rapid History of .NET


2002
.NET Framework
Windows based development



2016
.NET Core
Cross platform, open source
from Microsoft



2004
Mono Project, later Xamarin
First cross platform .NET option



2016/2017
.NET Standard
Specification for sharing code
across .NET



Challenges for Understanding the .NET Platform



.NET Developer



.NET Framework

.NET Core

Packages

Compatibility

Data Access

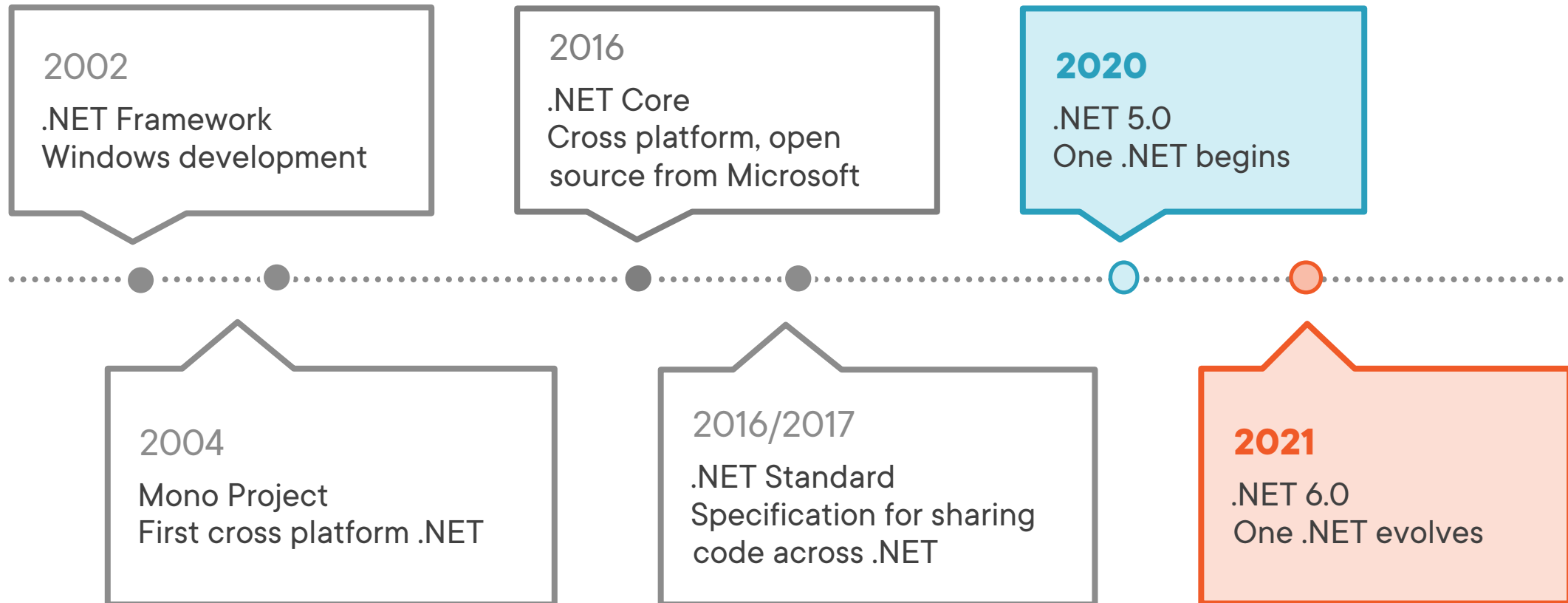
Runtimes

Tooling

Cloud integration



A Rapid History of .NET



Introducing .NET 6.0

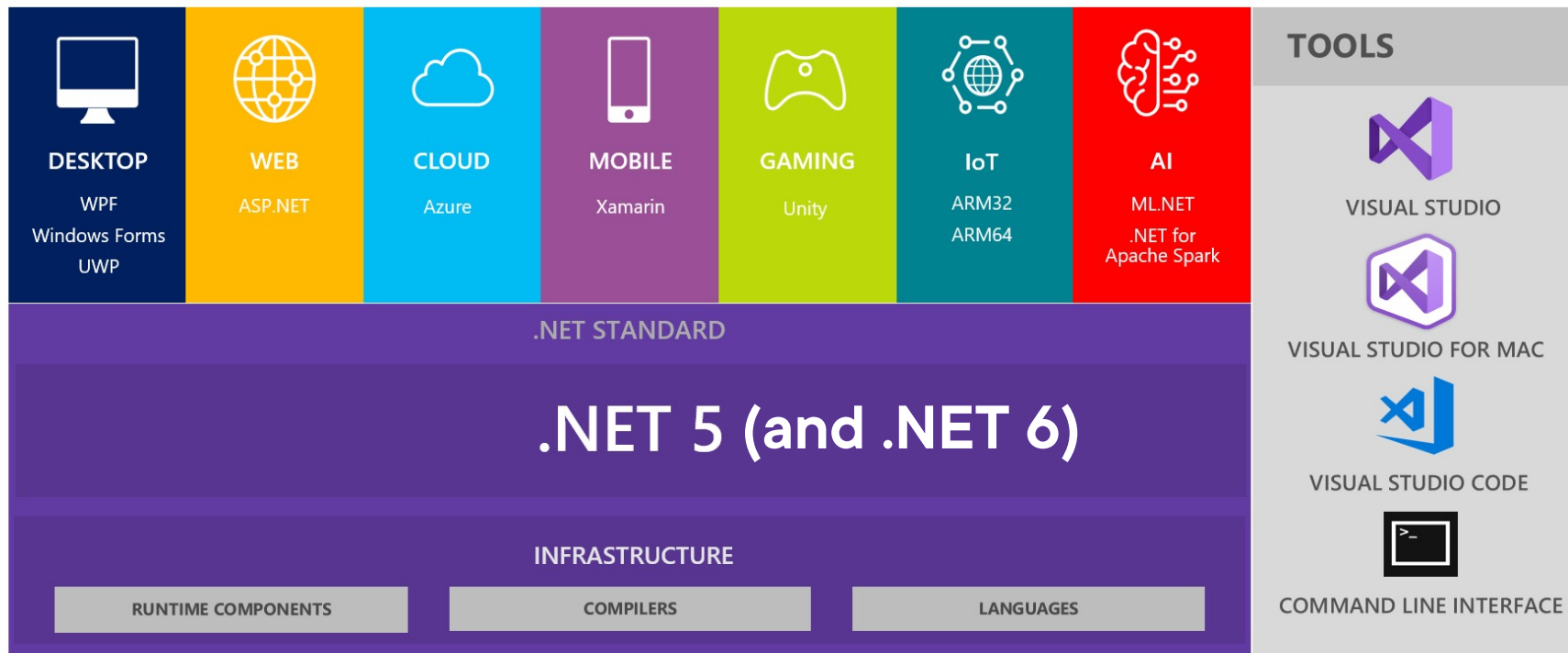


One .NET

A unified .NET platform with a single set of Base Class Libraries, SDK, languages, and tools for cross platform development.



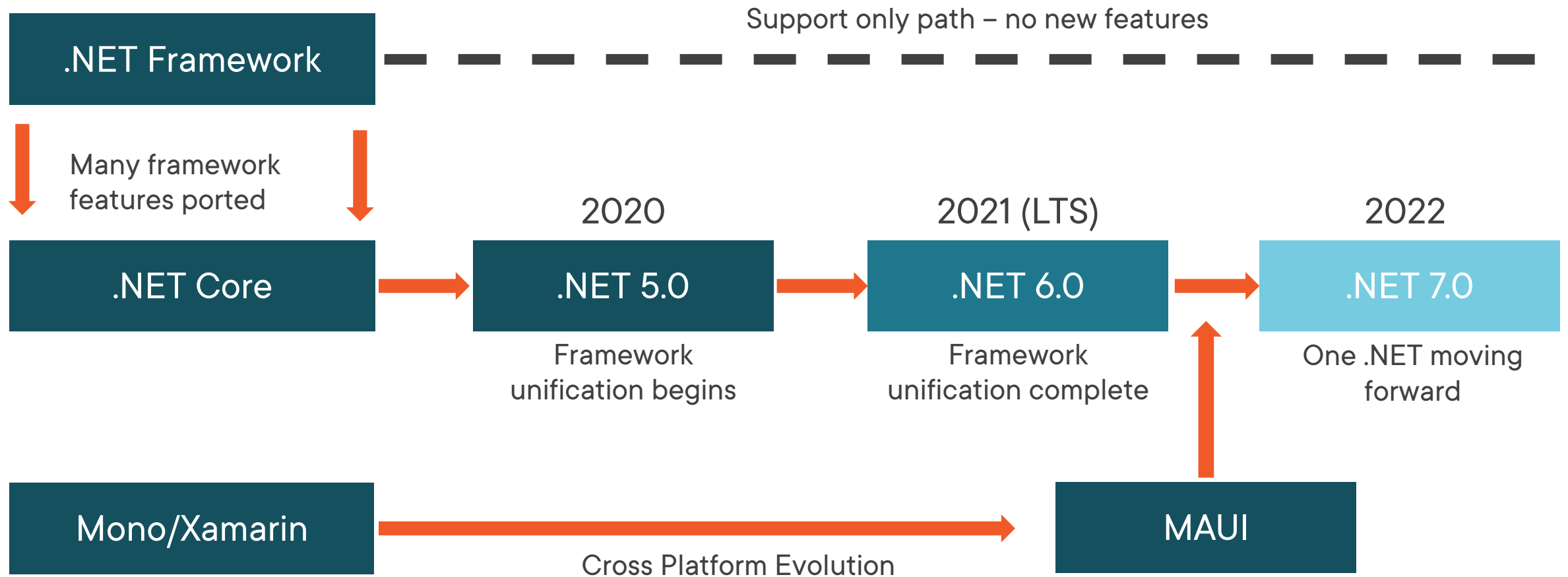
Visualizing a Unified .NET Platform



<https://devblogs.microsoft.com/dotnet/announcing-net-6/>



The Future of One .NET

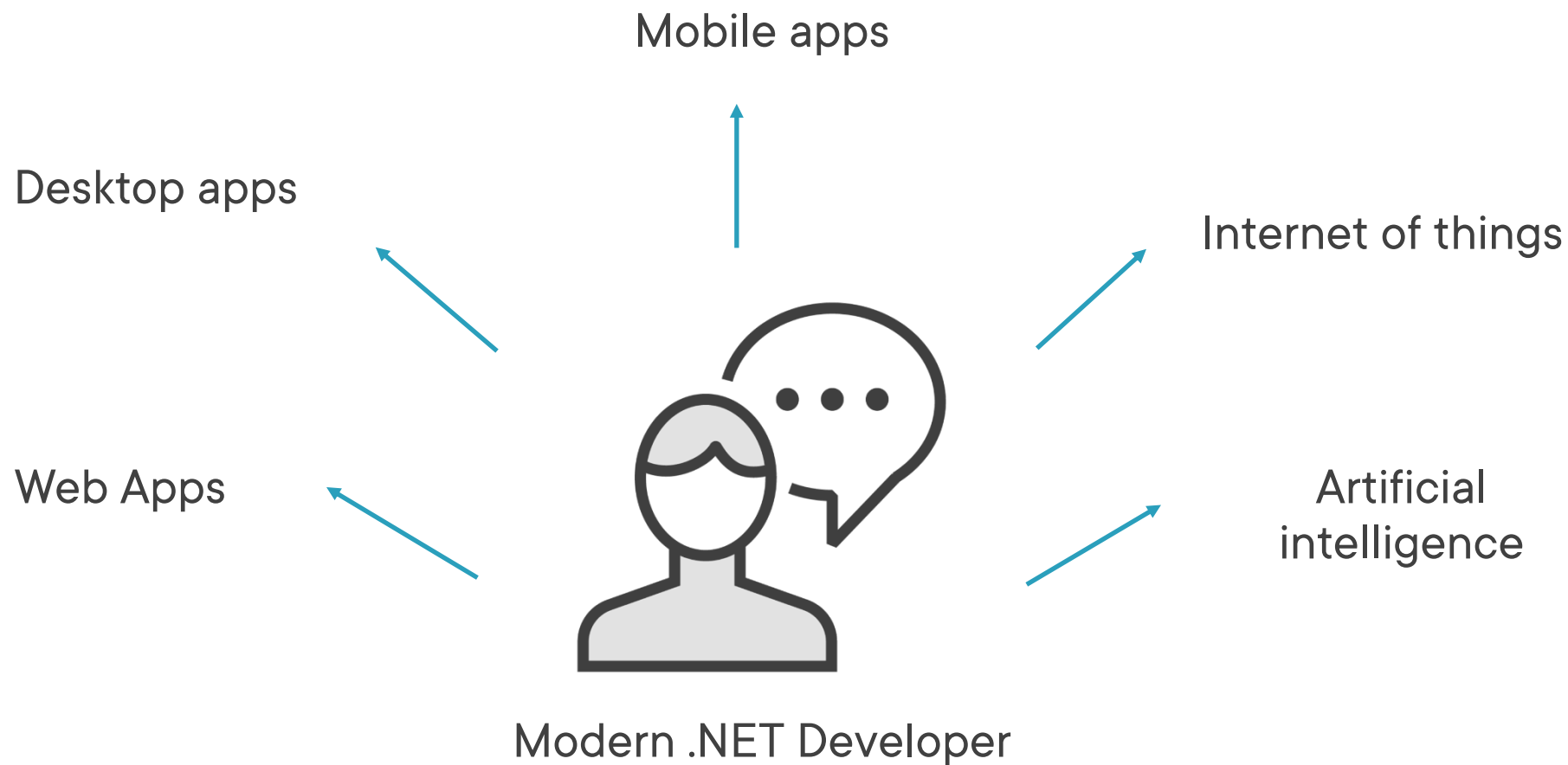


Understanding .NET Versioning

| Framework | Major Versions | | | | | | |
|----------------|----------------|-----|-----|-----|-----|-----|-----|
| One .NET | X | X | X | X | 5.0 | 6.0 | 7.0 |
| .NET Core | 1.0 | 2.0 | 3.0 | X | X | X | X |
| .NET Framework | 1.0 | 2.0 | 3.0 | 4.0 | X | X | X |



New Horizons for Existing Developers



Highlights of .NET 6.0

Unified Platform

Improved performance

C# 10

Hot reload

Improved build features

Significant new ASP.NET features

Visual Studio 2022



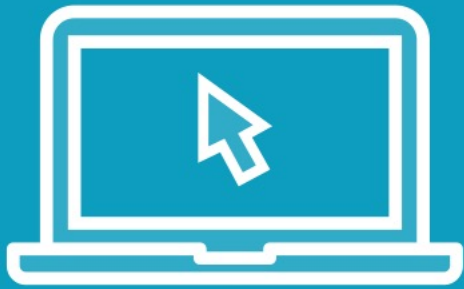
Focusing on ASP.NET in .NET 6.0



<https://devblogs.microsoft.com/dotnet/announcing-net-6/>



Demo



Installing .NET 6.0



.NET 6.0 Installation Options

Visual Studio 2022

**.NET 6.0 comes bundled with
Visual Studio 2022**

Standalone Install

**Download and .NET 6.0
directly by itself**



Exploring ASP.NET Core 6.0

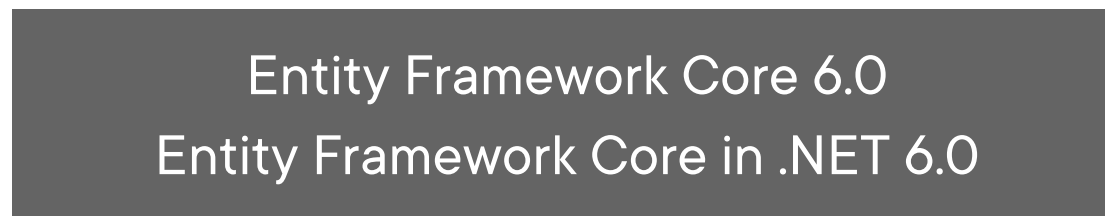
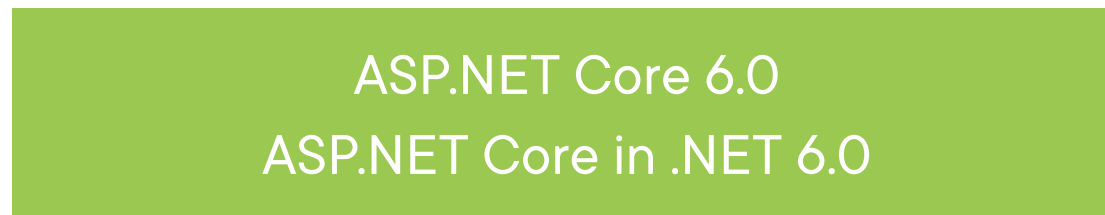


ASP.NET Naming Considerations

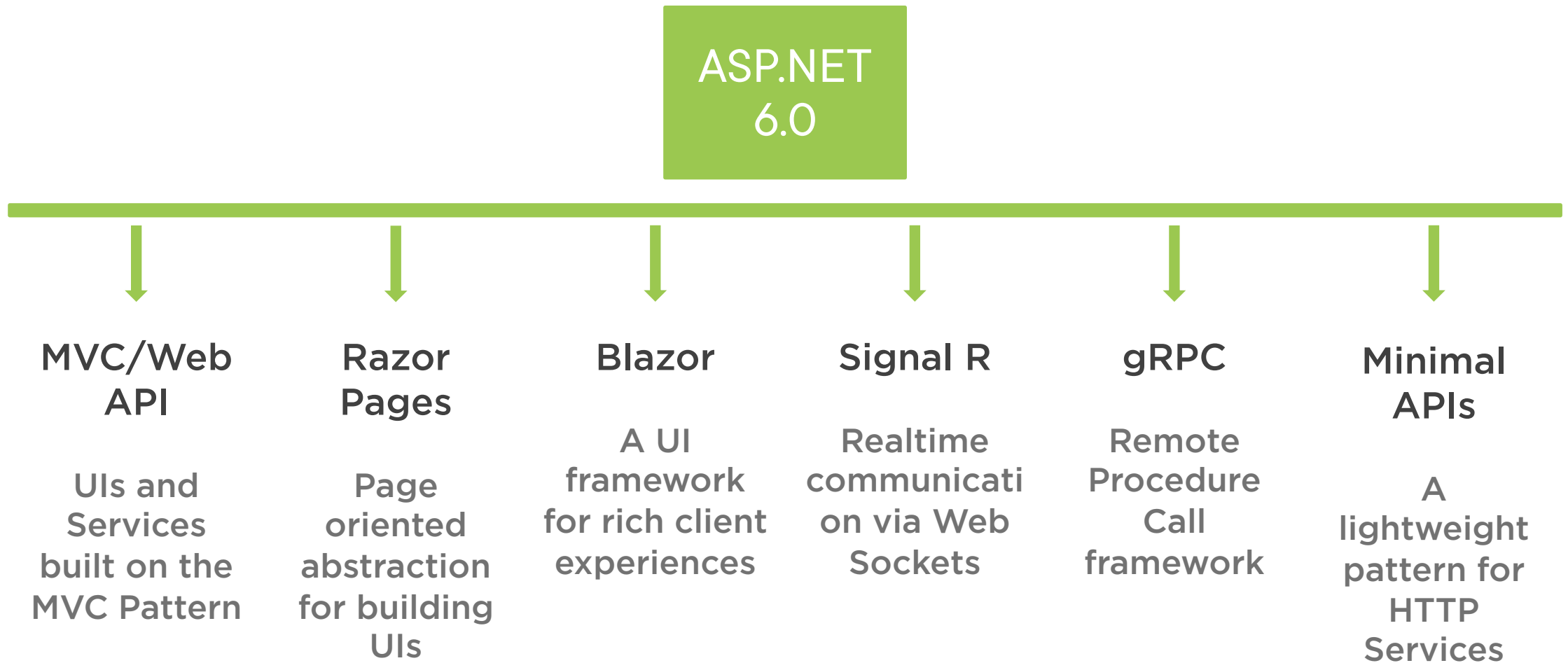
Overall platform and SDK



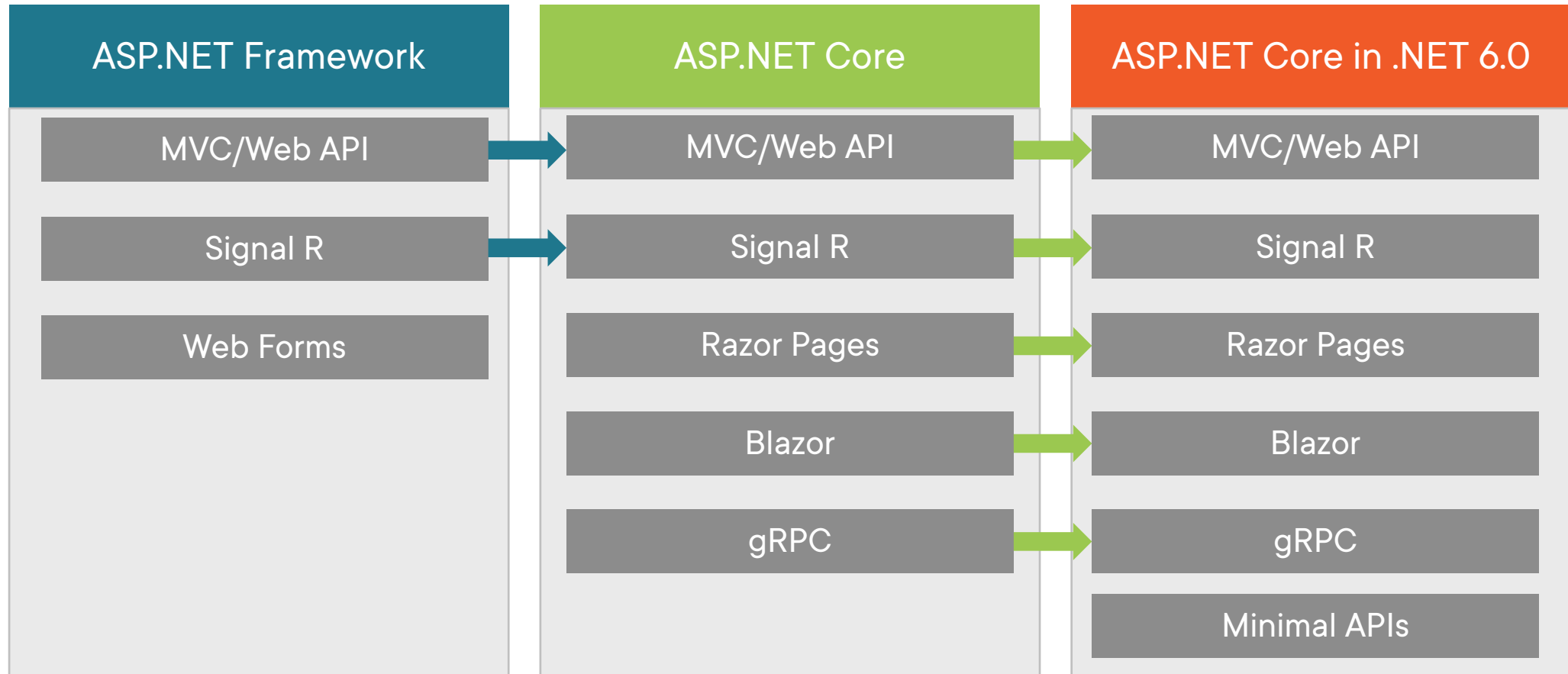
Specific application model names



The Application Patterns of ASP.NET Core 6.0



The Journey to ASP.NET Core 6.0



Categorizing ASP.NET Enhancements

MVC & Web API, Razor Pages, ASP.NET Platform

Minor new application features, but major new platform features

Minimal APIs

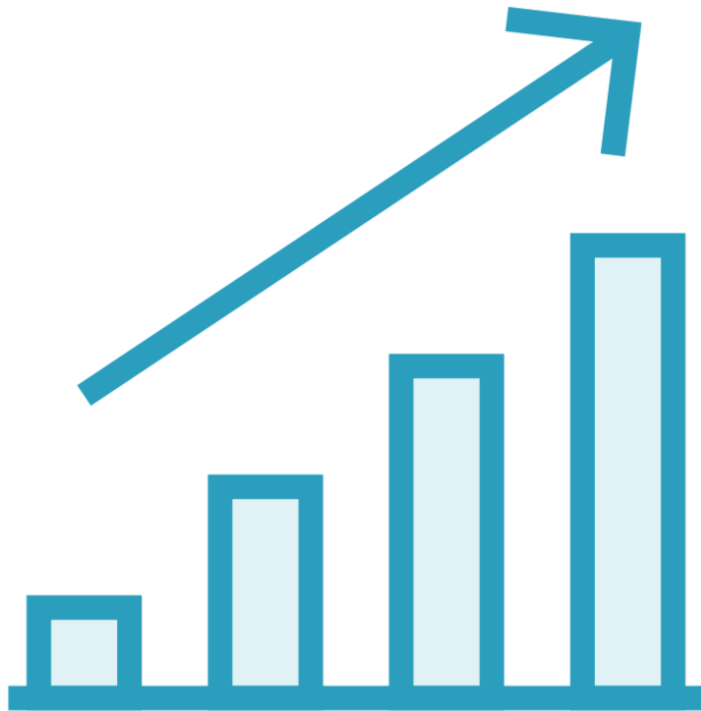
A significant new application pattern

Blazor

New quality of life and performance features



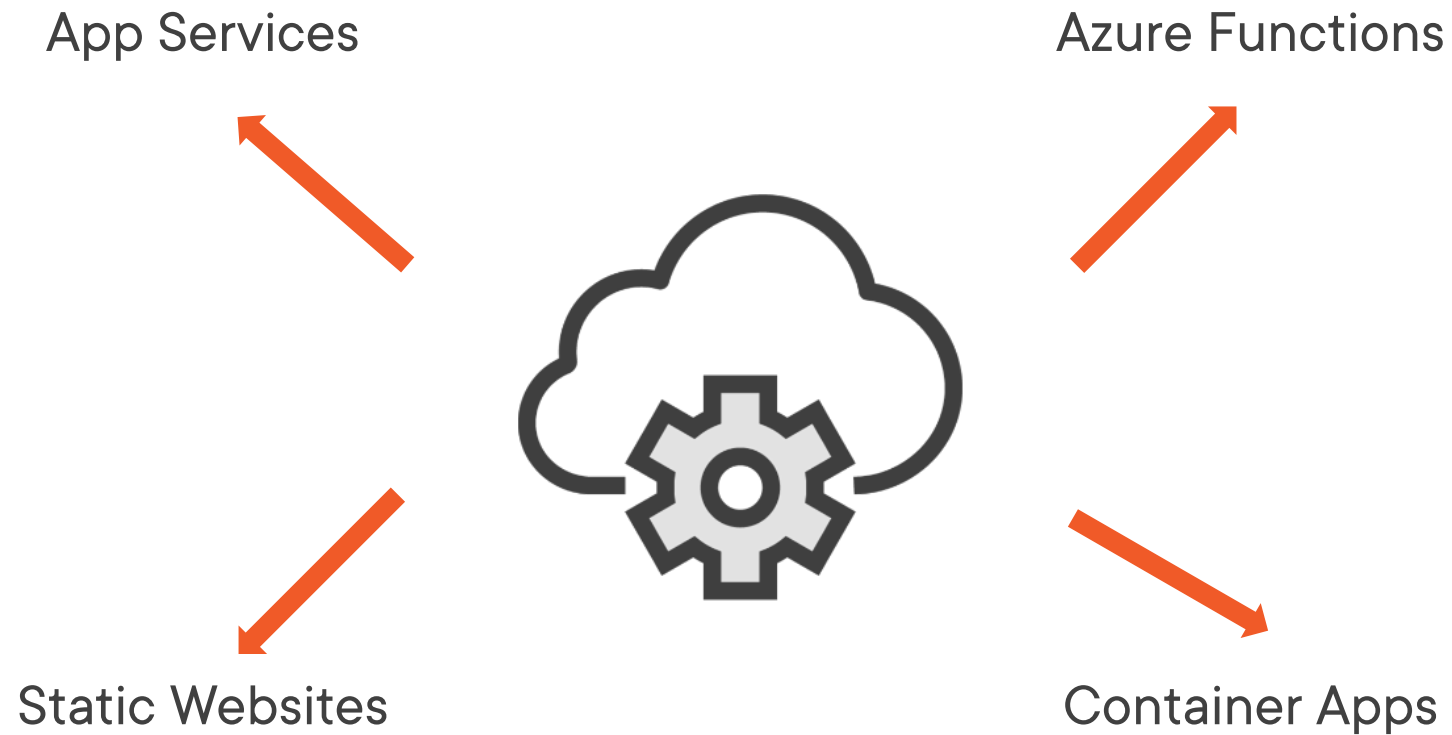
Performance Improvements in ASP.NET Core 6.0



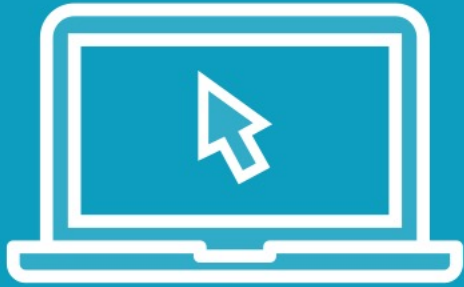
Performance improvements across all of .NET simply by upgrading



Azure Support for ASP.NET 6.0



Demo



Setting up the sample project





Introducing the Wired Brain Coffee app

- Online ordering
- Contact forms
- Promotions & info
- Expanding into admin features
- Dev team would like to utilize the latest features of .NET 6.0

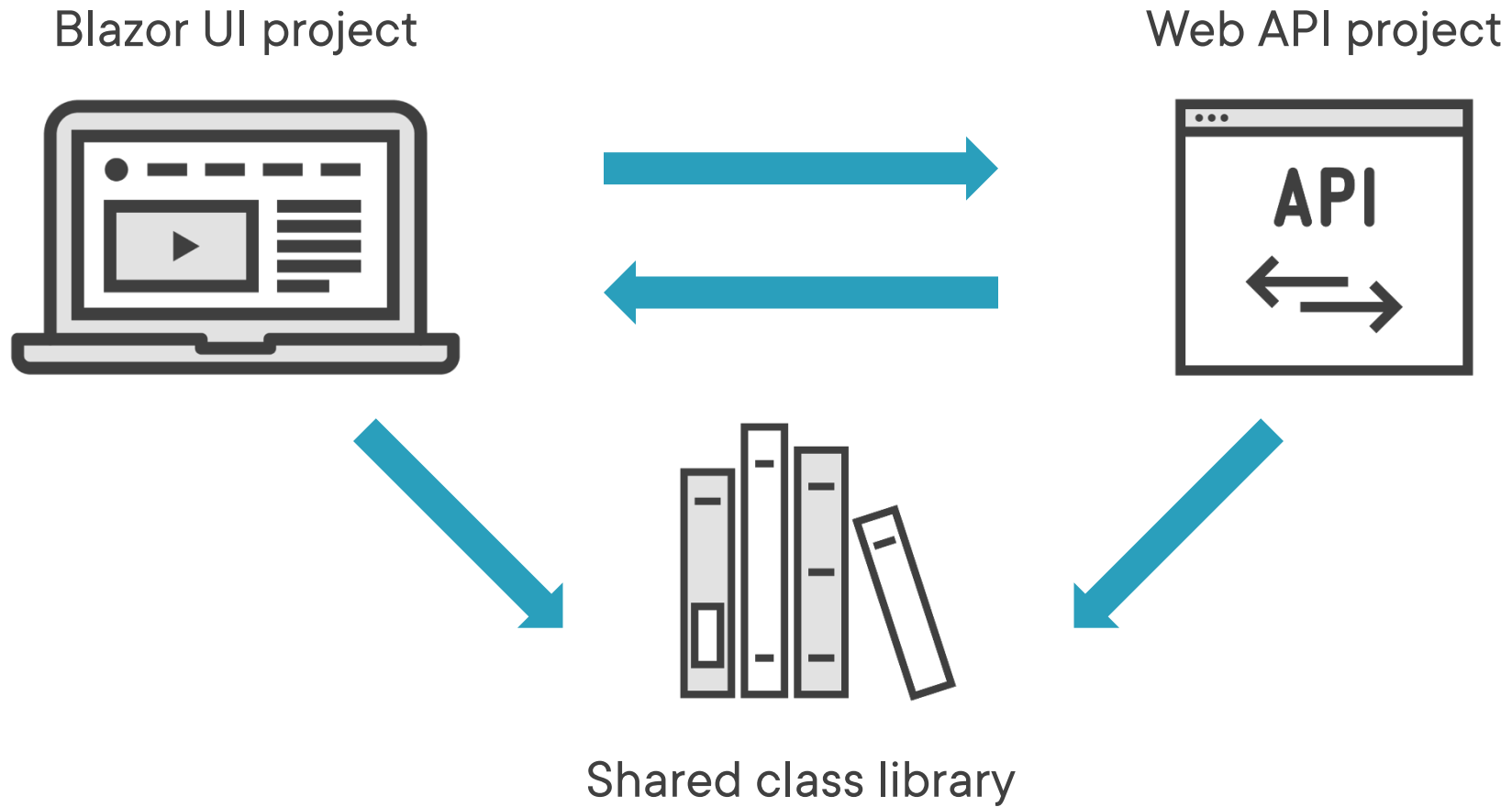


GitHub Links

<https://github.com/alex-wolf-ps/whats-new-aspdotnet-6>



The Wired Brain Coffee Architecture



Upgrade and Version Considerations



Upgrading to .NET 6.0 is painless for projects built on recent versions.



Effort Required to Migrate to ASP.NET Core 6.0



| Version | ASP.NET Core 2.0– ASP.NET 5.0 | ASP.NET Core 1.x | ASP.NET Framework |
|--------------------|----------------------------------|------------------|-------------------|
| Level of Effort | | | |



Upgrading to ASP.NET Core 6.0

The more recent your ASP.NET Core version, the easier it will be.

Filename.code

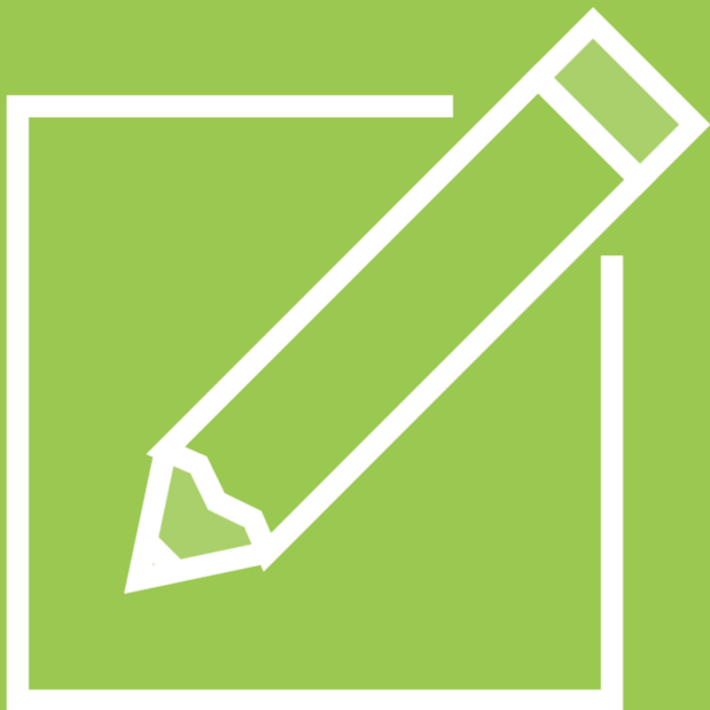
```
<Project Sdk="Microsoft.NET.Sdk.BlazorWebAssembly">  
  <PropertyGroup>  
    <TargetFramework>net6.0</TargetFramework>  
  </PropertyGroup>  
  
  <PackageReference include="System.Net.Http.Json  
version="6.0">  
  
</Project>
```

In some cases, update project SDK

Update target framework

Update packages

Revisit startup and program files



A Quick Note

Most structural project changes in .NET 6.0 are entirely optional



A note about .NET Standard.



Guidelines for .NET Packages Going Forward



Target [.NET 5.0](#) to support only .NET 5.0 or higher
Target [.NET 6.0](#) to support only .NET 6.0 or higher
And so on....



Target [.NET Standard 2.1](#) to share code between .NET 5.0+, .NET Core 3+, recent versions of Xamarin, but *not* .NET Framework



Use [.NET Standard 2.0](#) to share code between .NET Framework and all other implementations



Version Specific Upgrade Details

App upgrades:

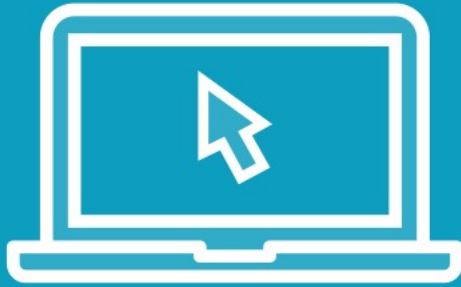
<https://docs.microsoft.com/en-us/aspnet/core/migration>

Package upgrades:

<https://docs.microsoft.com/en-us/dotnet/standard/net-standard>



Demo



Updating an Existing Project to .NET 6.0



Sample Upgrade Project

<https://github.com/alex-wolf-ps/aspdotnet5-upgrade>



Summary



.NET 6.0 provides a unified set of Base Class Libraries, SDKs, languages, and tools

.NET 6.0 arrives with C# 10, performance improvements, VS 2022, Hot Reload, and more

ASP.NET Core 6.0 introduces plenty of new platform and tooling enhancements

Minimal APIs deliver a major new web service pattern for ASP.NET Core 6.0

Blazor has received many new quality of life and performance improvements

Upgrading to ASP.NET Core 6.0 from recent versions is generally approachable and painless

