Essential .NET Standard 2.0

Mark Michaelis

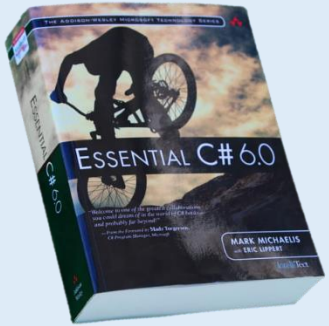
IntelliTect.com | info@intellitect.com

IntelliTect

About Us

Mark Michaelis

Chief Technical Architect,
Author, Trainer



Microsoft® Most Valuable Professional

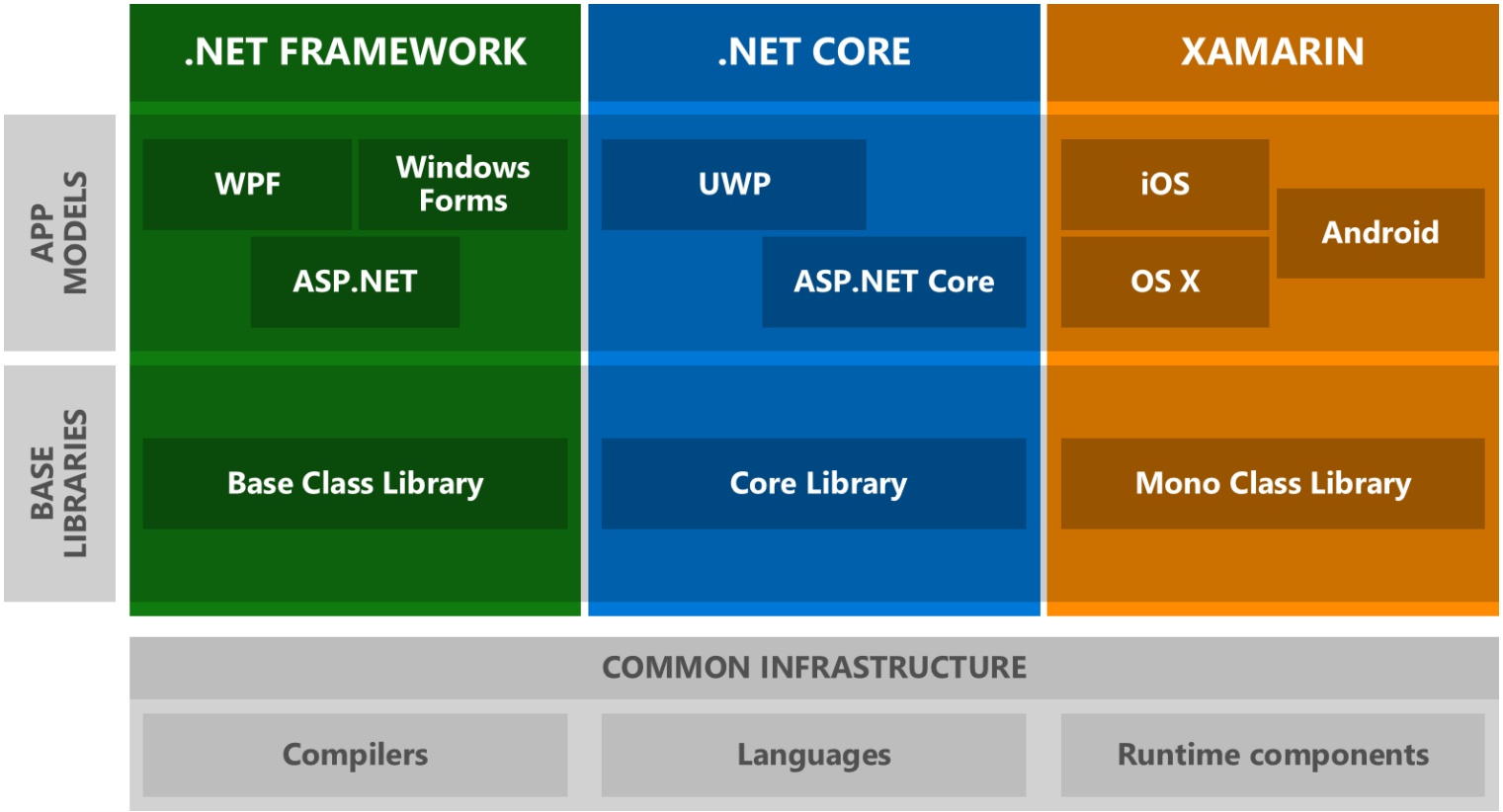
Microsoft® Regional Director

Professional Scrum Master II

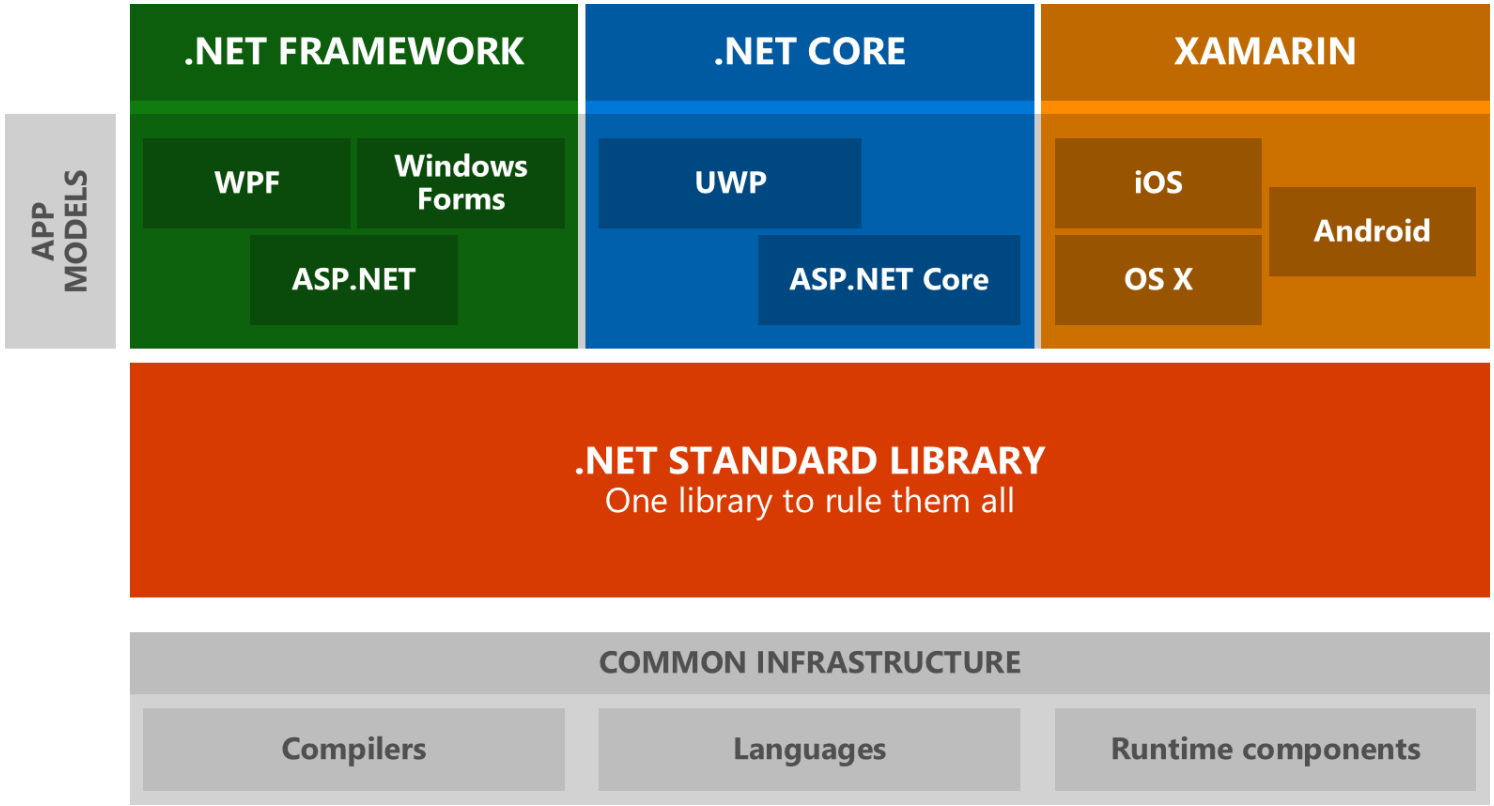
Before .NET Standard 2.0

- Portability is relatively limited because .NET Core is relatively small
- .NET Framework libraries (the vast number of libraries that exist today) are only available on Windows
- .NET Core 1.1 is based on a large number of mini-assemblies, thus creating a pain to manage and a pain to upgrade.

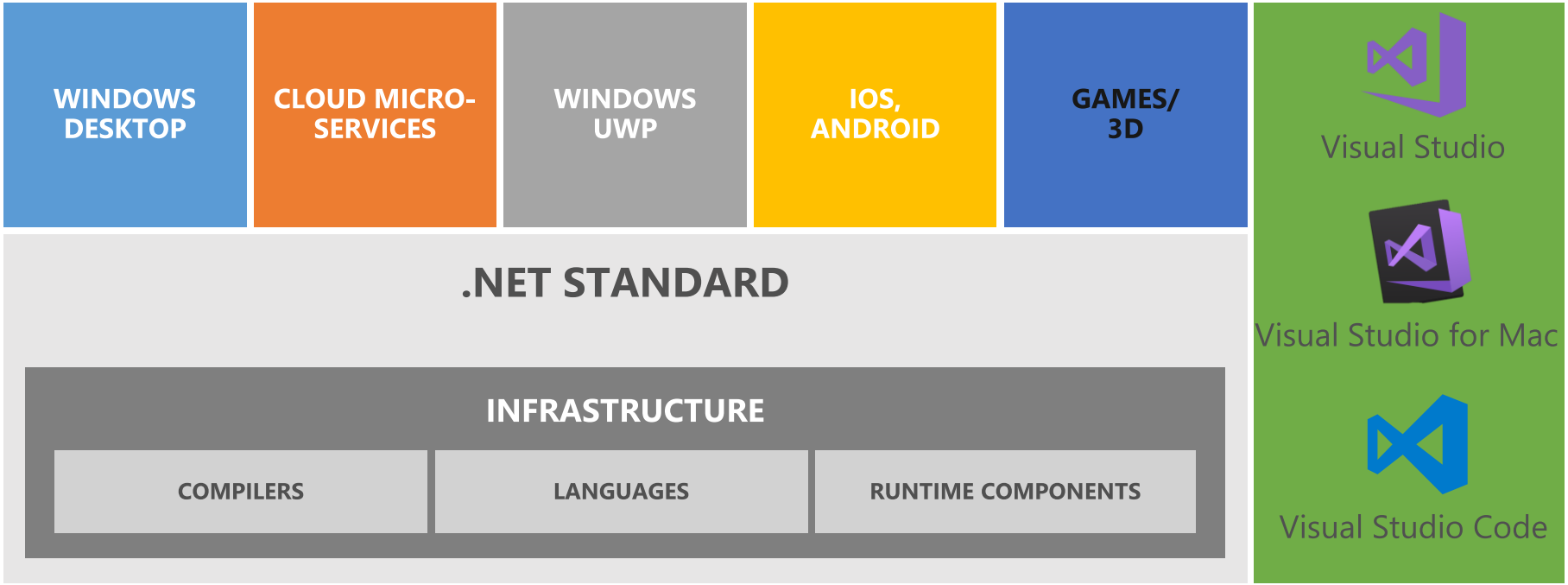
.NET Runtimes Independent of a Standard



.NET Runtimes with .NET Standard



.NET Standard: a Common API for Platform Providers

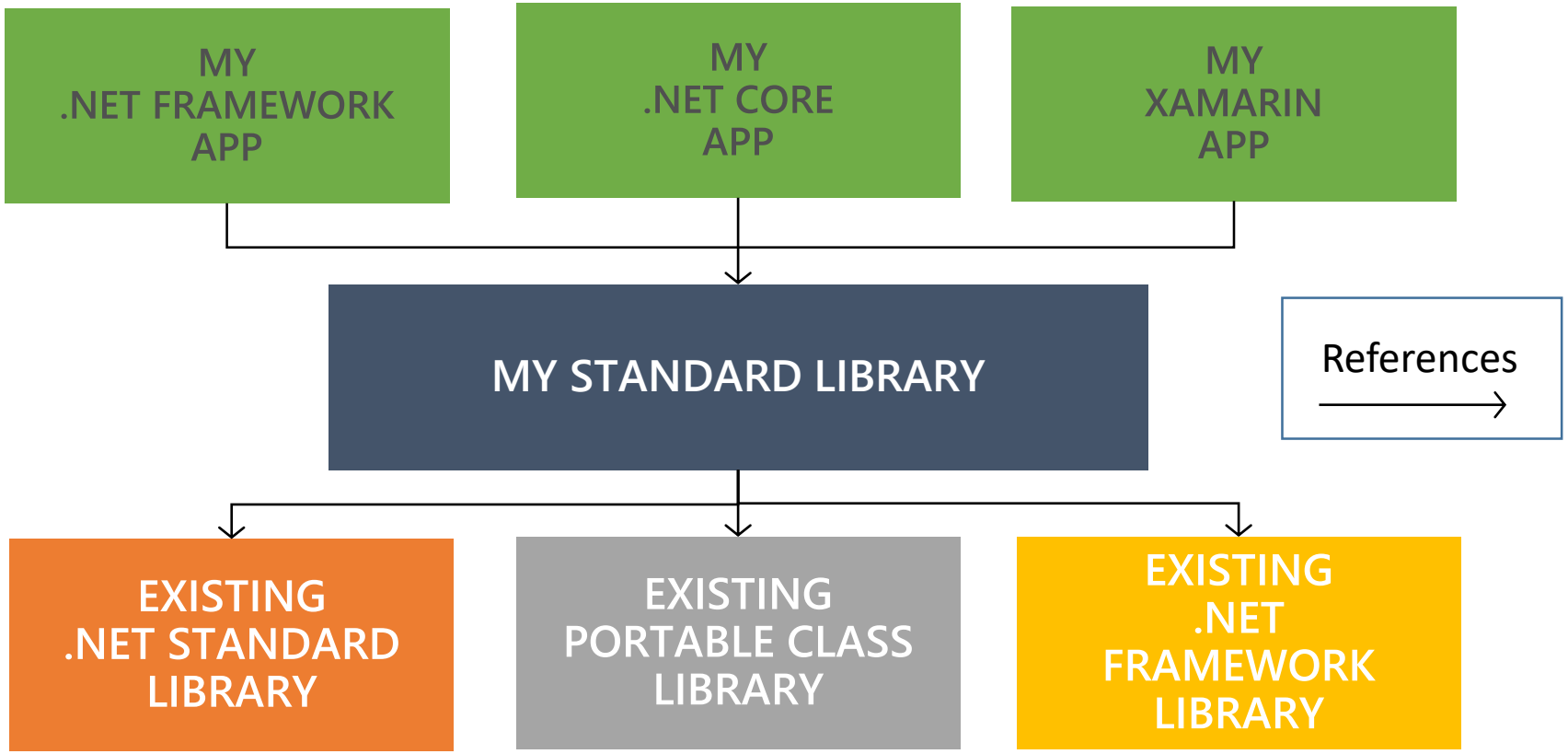


What is .NET Standard

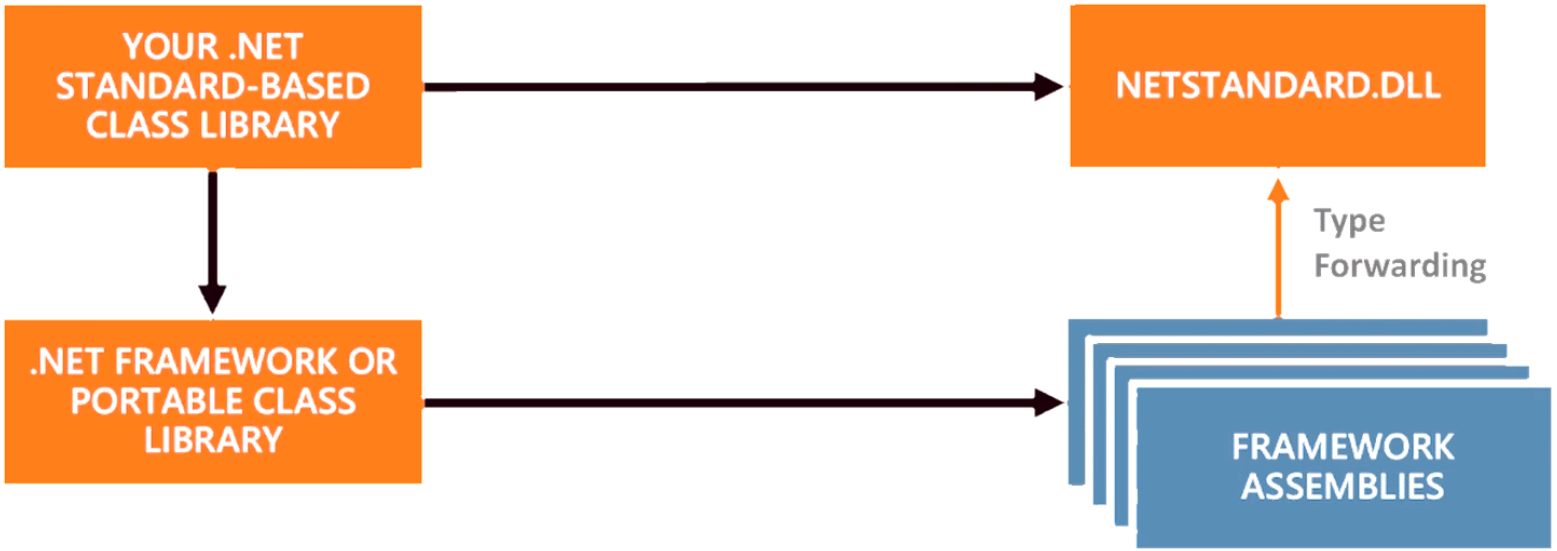
.NET Standard is a set of APIs that all .NET platforms have to implement

Libraries can reference a .NET Standard in order to be compatible on platforms that support that standard

Any .NET Standard App Using Any .NET Standard Library

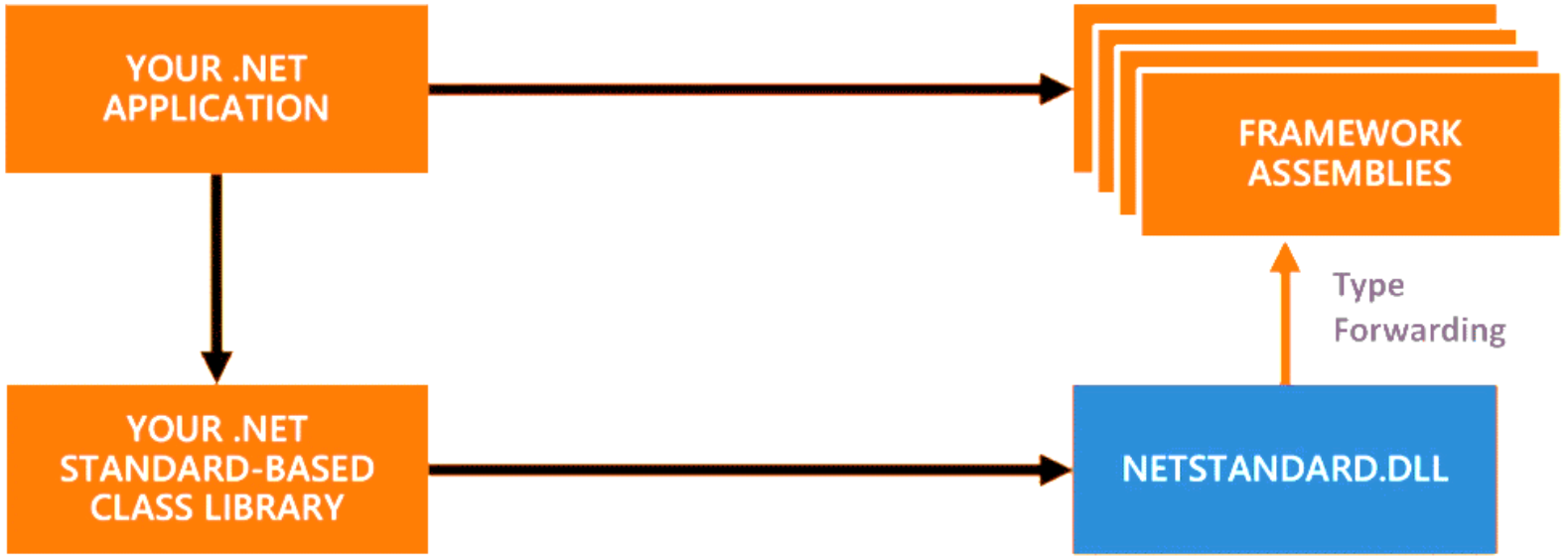


Dotnet Standard application referencing



Non-Dotnet Standard Assemblies

Non-Dotnet Standard referencing



Dotnet Standard Assemblies

.NET Standard Summary

Libraries targeting a particular version of .NET Standard will successfully run on the platform that implements that same version or higher

How .NET Standard 2.0 Works

.NET Standard is implemented in the NET Standard.Library nupkg which contains the netstandard.dll assembly.

Each platform has its own implementation of netstandard.dll

At runtime, .NET Standard uses type forwarding to enable applications to access .NET Framework and PCL assemblies.

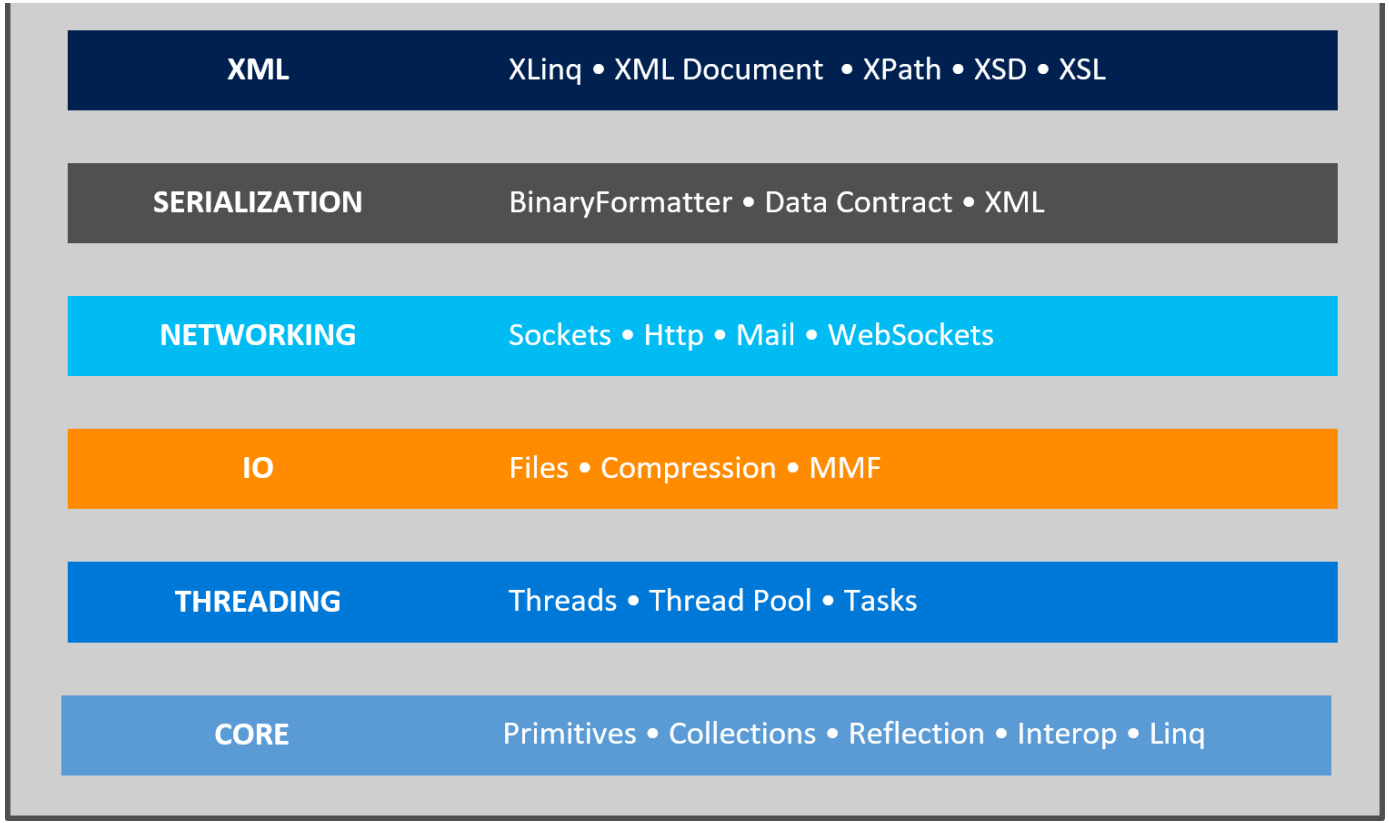
At build time, applications that only use .NET Standard API only reference the netstandard.dll

.NET Standard Library Advantages

- Provides a specification for any platform to implement
- A consistent API across all frameworks
- Enabling the reuse of code across frameworks
- Dotnet.exe runs restore implicitly
- Providing the ability to consume third-party libraries across frameworks
- All .NET runtimes provided by Microsoft will implement the standard

What's Added to .NET Standard in 2.0

Add Support for .NET Framework BCL APIs



What won't work?

XML, Serialization, Networking, IO, Threading, XML, Reflection

WPF

Xamarin Forms

Direct Operating System API access

etc

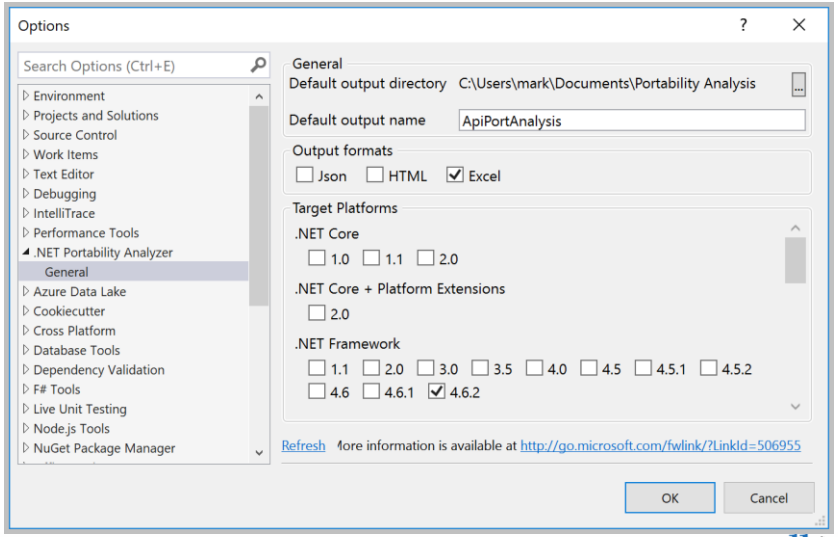
.NET Standard 2.0

Outside .NET Standard 2.0

Use Determining What is Supported

- .NET Standard 2.0 adds **14,994 APIs** that .NET Framework 4.6.1 already supports
- .NET Standard 2.0 only has **43 APIs** that .NET Framework 4.6.1 doesn't support

- Use .NET Portability Analyzer (apiport.exe) to verify application compatibility with .NET Standard



IntelliTect

Miscellaneous

- .NET Native for .NET Core is still missing but targeted for next year
- .NET Native support for .NET Standard is targeted for the fall
- UWP will support .NET Standard with the Windows Creator Update

Summary

CONSIDER migrating .NET Framework libraries if supporting more .NET Platforms is compelling

DO NOT change your application to support .NET Standard just for the sake of it

DO use .NET Standard in favor of .NET Portable Class Libraries (PCL)

DO pay attention to overlapping API: ie. System.Configuration & Microsoft.Extensions.Configuration

Additional Resources

- **Installing .NET Standard 2.0**
  https://www.microsoft.com/net/core/preview
- **Introducing .NET Standard**
  https://blogs.msdn.microsoft.com/dotnet/2016/09/26/introducing-net-standard/
- **Three Runtimes, one standard… .NET Standard: All in Visual Studio 2017**
  https://channel9.msdn.com/events/Build/2017/B8001
- **.NET Standard 2.0 and .NET Core 2.0**
  https://channel9.msdn.com/events/Build/2017/C9L18

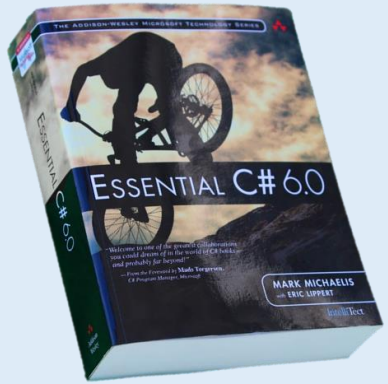
About Us

Mark Michaelis

Chief Technical Architect,
Author, Trainer

mark@IntelliTect.com

Twitter: @MarkMichaelis,
fb.com/MarkMichaelis



Microsoft® Most Valuable Professional

Microsoft® Regional Director

Professional Scrum Master II