

Benvenuti!

Cristian Civera

Microsoft Azure MVP

cristian@cristiancivera.com | @cristiancivera

 aspitalia.com

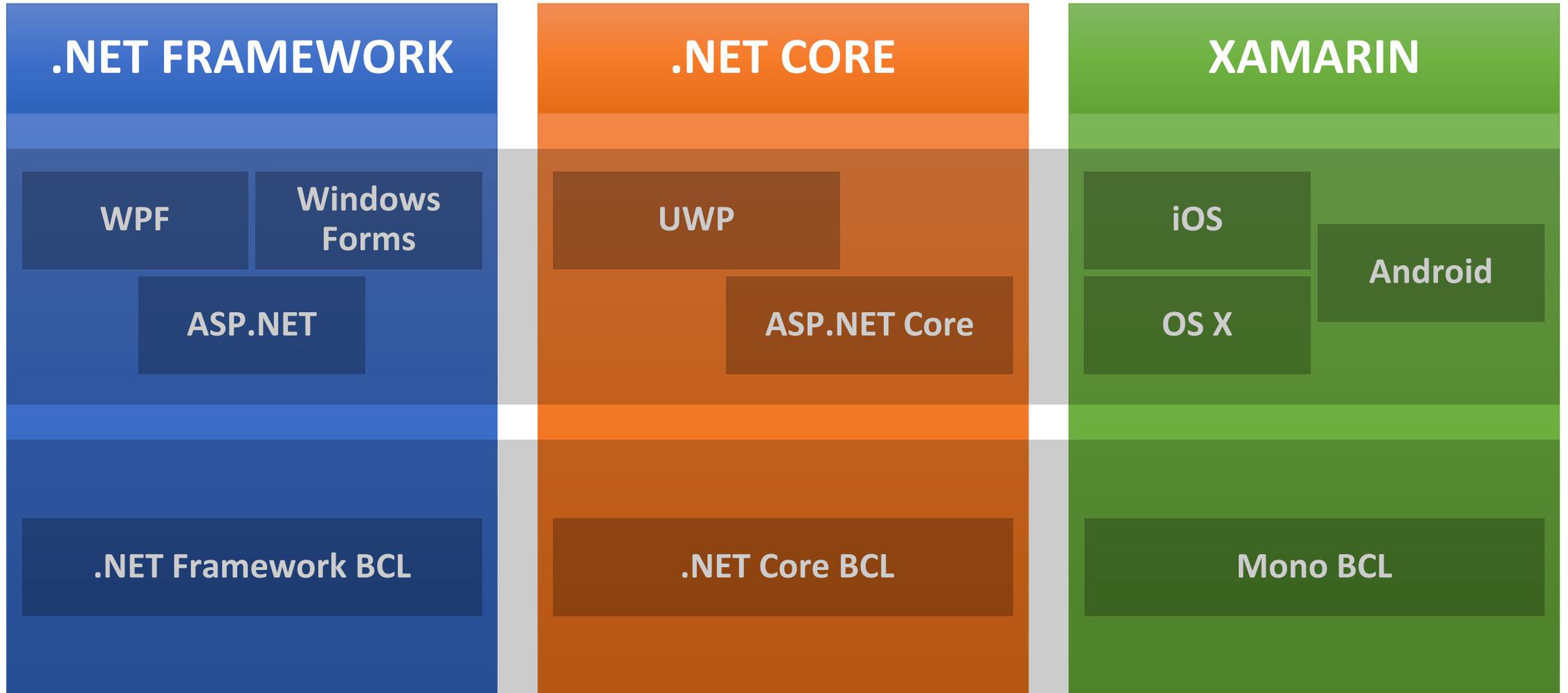


Agenda

- Storia di .NET
- .NET Core
- .NET Standard

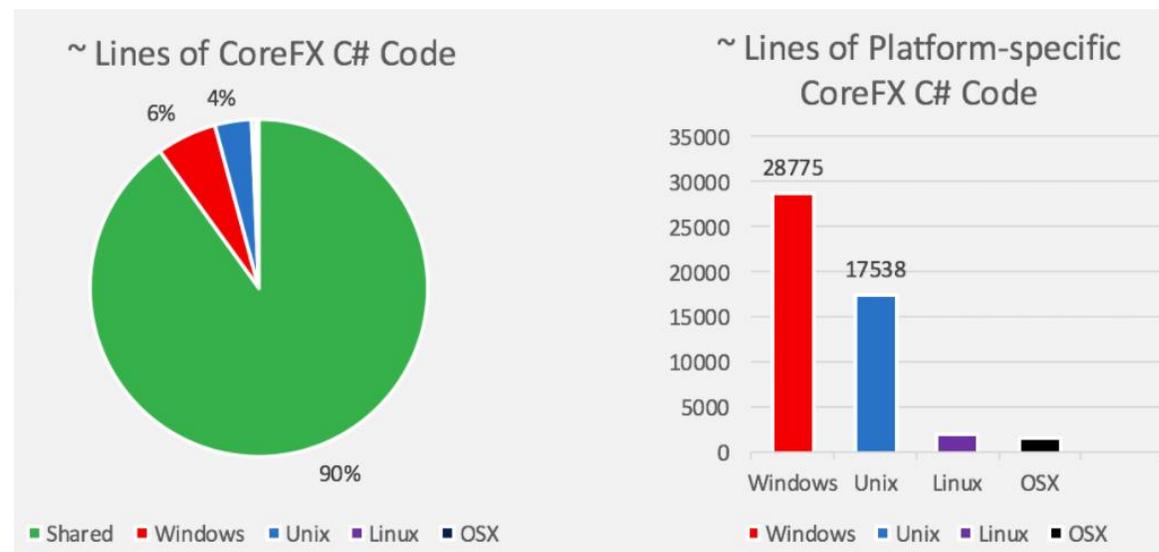


.NET ecosystem



.NET Core

- Open source
 - 1k PR al mese
 - .NET Core 2: performance migliorate su collezioni, LINQ, serializzazione, IO
- Runtime cross platform
 - Sviluppo parallelo sui vari OS
 - X86, x64, ARM
 - Portable pdb
- DotNetCli
 - Estensioni
- Basato su NuGet



.NET Core

- Estrema modularità
 - Namespace ~ Assembly
 - IoC first citizen
- xcopy deploy
- Librerie
 - ASP.NET Core, Entity Framework Core
- Rilasci
 - Long Term Support (LTS): 3y + 1y
 - Current releases: 3m + 1y

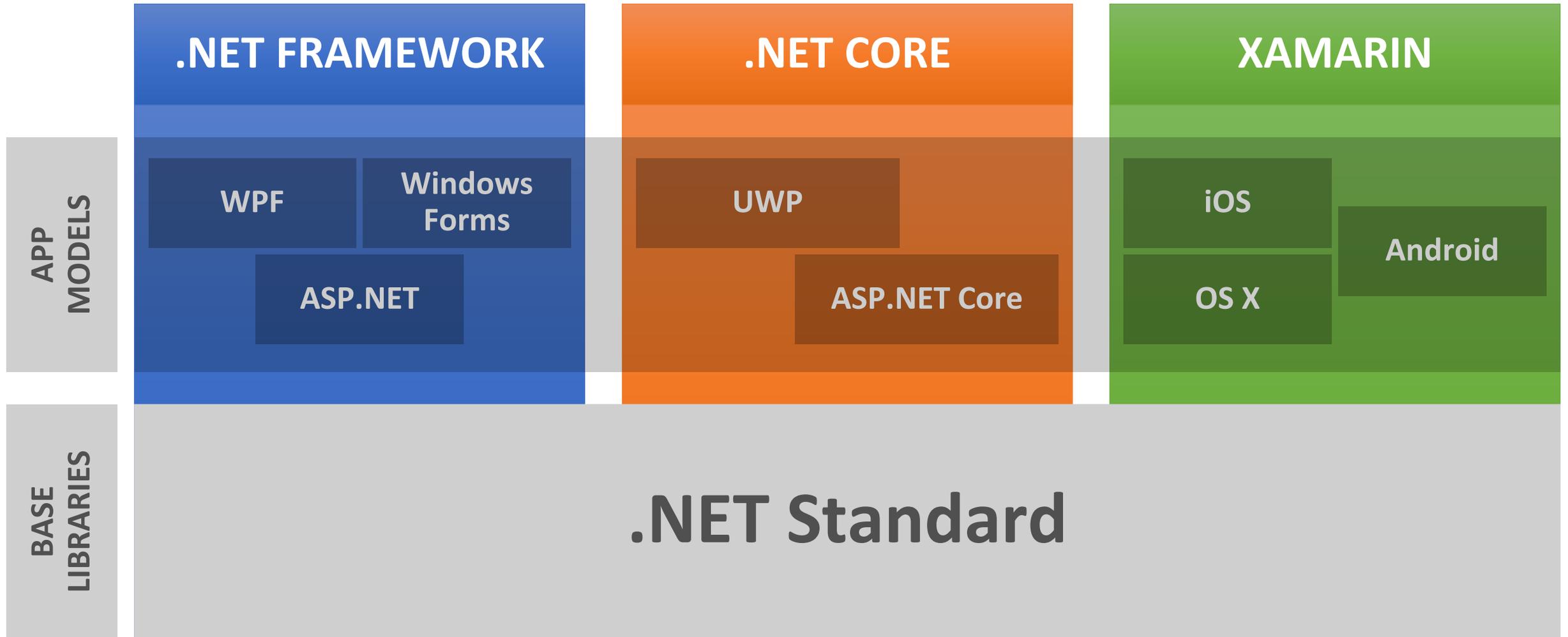
New VS2017 csproj type

- Generazione diretta di package
 - Referenze basate su NuGet
- Più leggero
- Modifica diretta
- Niente più packages.config
- File inclusi per default
 - Migliore integrazione con source control
- Non è un esclusiva di .NET Core
 - Alcune cose non vanno, es. class designer

NuGet

- V4.x: più veloce
 - Niente più automazioni
 - Integrato in msbuild
- Cache generale
 - Non più cartella packages
 - Direttamente in csproj
 - Restore automatico
- Packages e Metapackages
 - Gestione gerarchica

.NET - reusing code



.NET Standard Versions

- .NET Standard è una specifica
 - Set di API implementate dalle piattaforme
 - Versioni senza breaking changes
 - <http://immo.landwerth.net/netstandard-versions/>
- .NET Standard ~ Specifiche HTML
 - .NET Framework / Core / Xamarin ~ Browser
- .NET Standard != .NET Core
 - Una delle tre piattaforme
 - Molte package .NET Core in realtà sono .NET Standard

.NET Standard 2.0

- +20K APIs
 - Libreria ~ va su tutte le piattaforme
- .NET Core 2, Xamarin, UWP e .NET Framework 4.6.1
 - In realtà .NET Framework 4.7.1
- BCL ampliata
 - System.Drawing, System.Data, System.IO, System.Net, System.Xml, System.Runtime, System.Security, System.Transactions
- WCF: System.ServiceModel.*

.NET Standard: what's missing

- Non confondere le librerie con la BCL
 - Es. ASP.NET è on top
 - Quindi no System.Web, WinForm, WPF, WF
- No System.IO.Port, System.Device, System.ServiceProcess, System.CodeDom
 - Alternative, anche fornite da Microsoft
- A volte alcuni overload

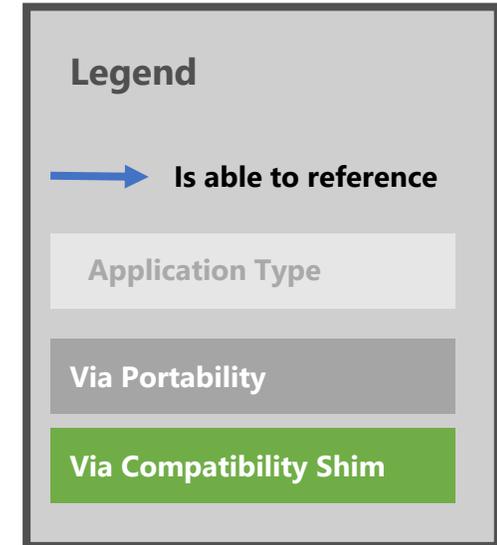
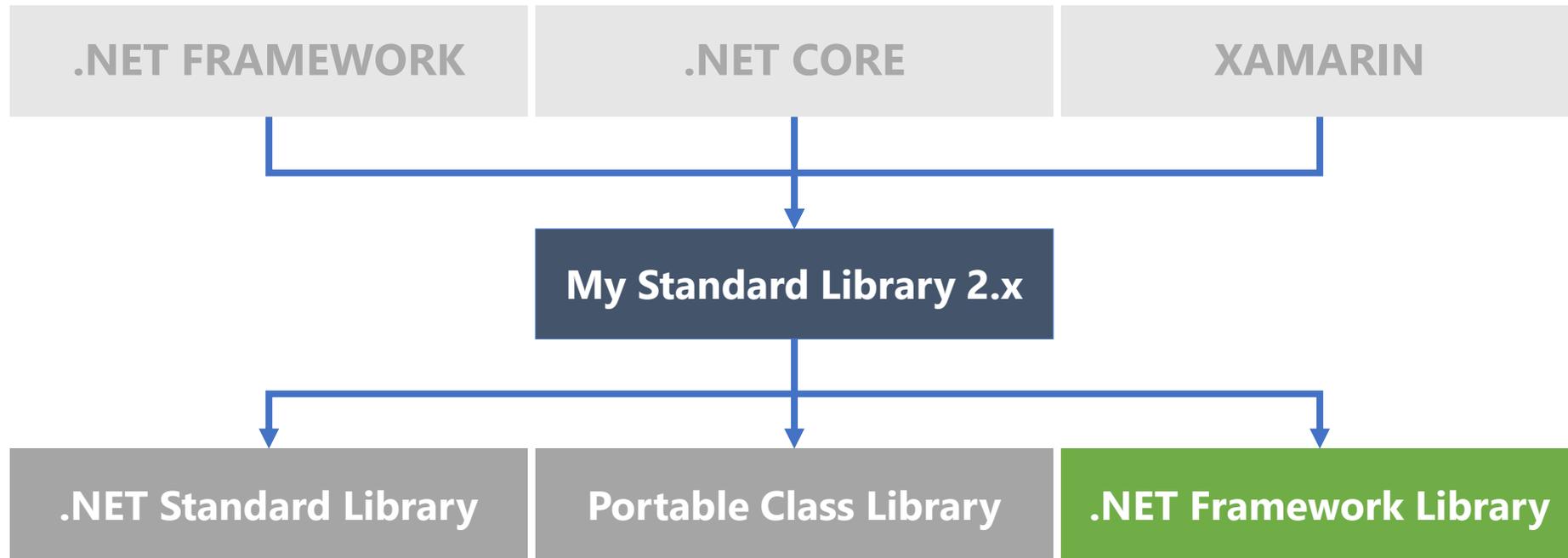
Developing .NET Standard libs

- Scegliere la versione standard più bassa
 - Netstandardx.y
 - ~ da netcoreappx.y
- Evitare la compilazione multi framework
 - Referenze condizionali e direttive di compilazione
- Abilitare la generazione NuGet
- Usare dipendenze .NET Standard
 - Referenze di progetto = dipendenza NuGet

Publish for a specific runtime

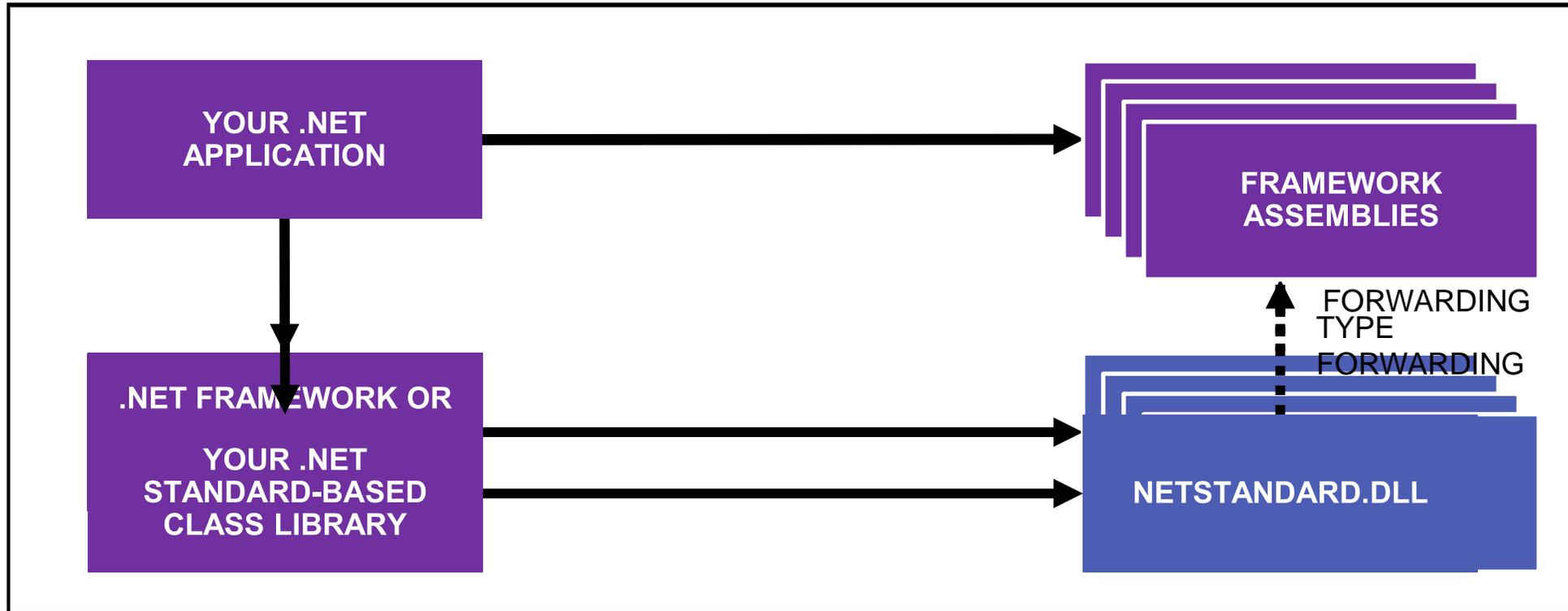
- Self-contained deployment
- `dotnet publish -c release -r win-x64`
 - linux-x64, osx-x64, android
 - È possibile scegliere la distro/os specifici
- Visual Studio 2017 pubblica solo framework-dependent

.NET assembly references



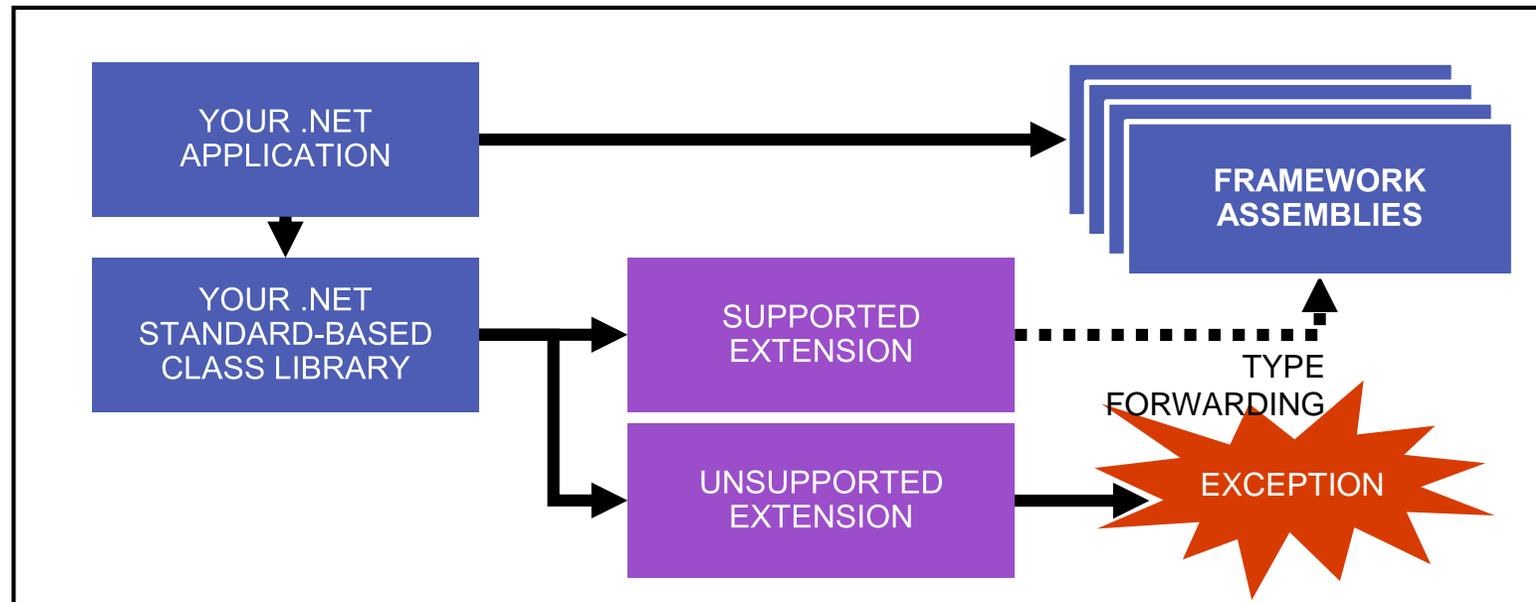
Reference .NET Standard assemblies

Arco di compilazione



Reference .NET Framework assemblies

- ~70% NuGet sono API compatibile
- Niente ricompilazione
 - Warning da Visual Studio
 - Errore a runtime se API mancante



.NET Future

- .NET Framework
 - Consolidamento 4.x
 - WinForm->WPF->UWP
- Xamarin
 - Compatibilità .NET Standard
- .NET Core 2.1: Q1 2018
- Altre librerie .NET verso .NET Standard
- Occhio ai WebAssembly!

Grazie!

@cristiancivera

cristian@cristiancivera.com

Materiale su

<http://aspit.co/netconf-17>

