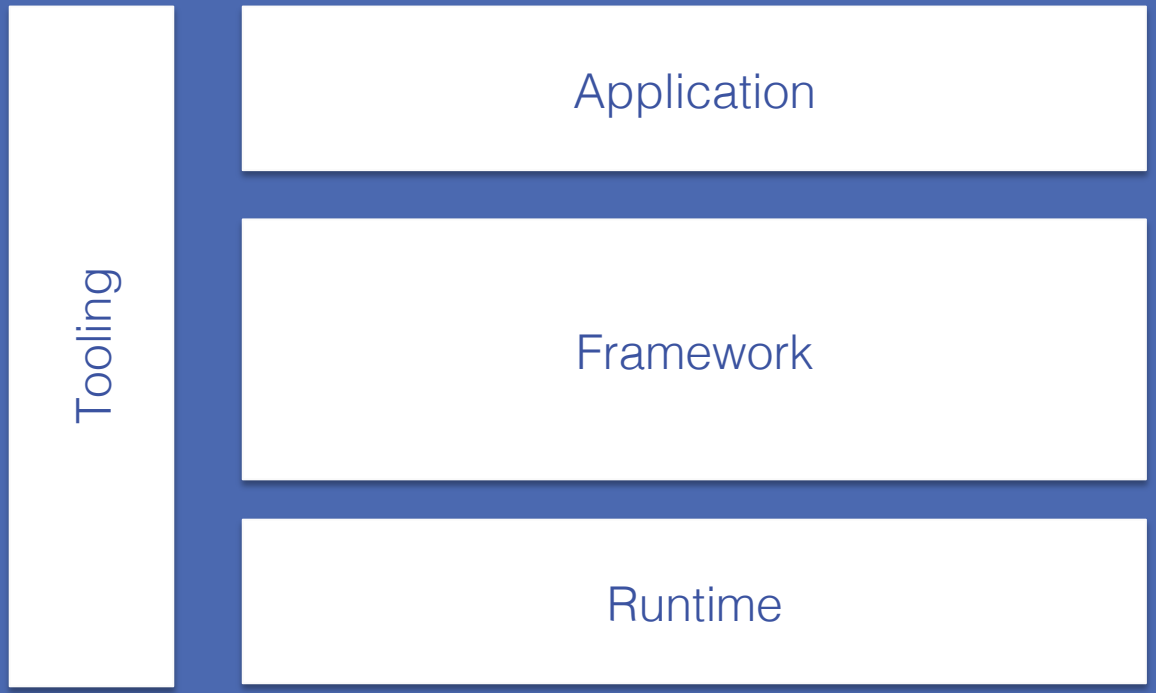
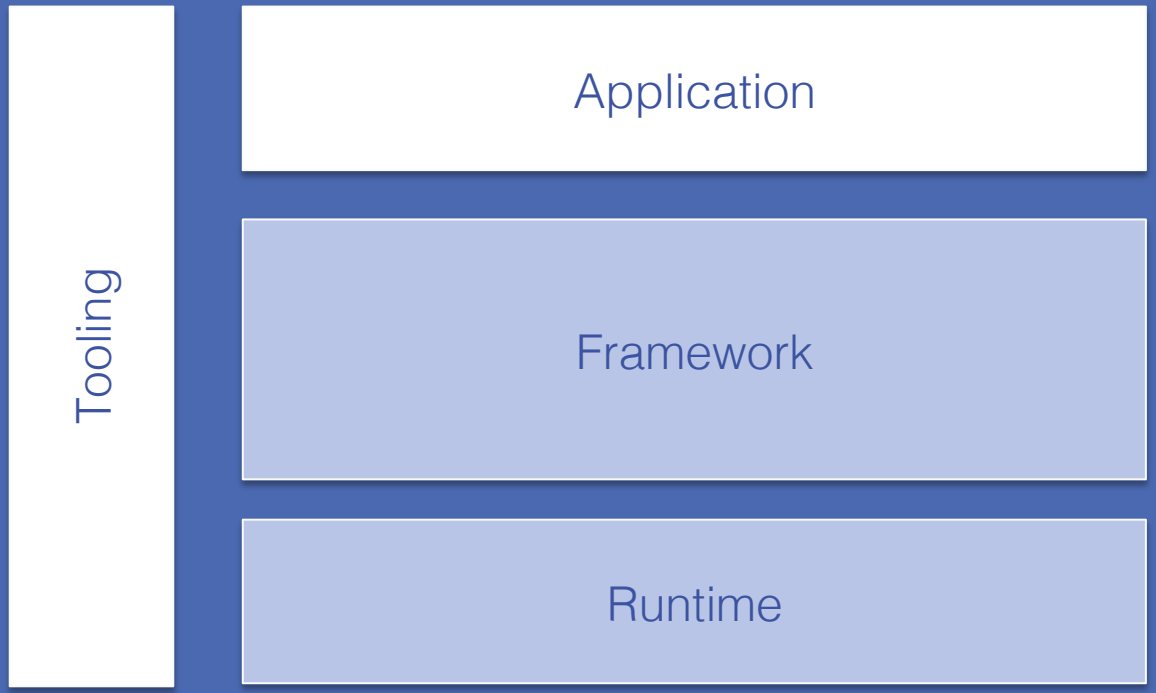
ASP.NET 5, .NET 4.6, and Visual Studio 2015

Nate McMaster
@natemcmaster

Overview



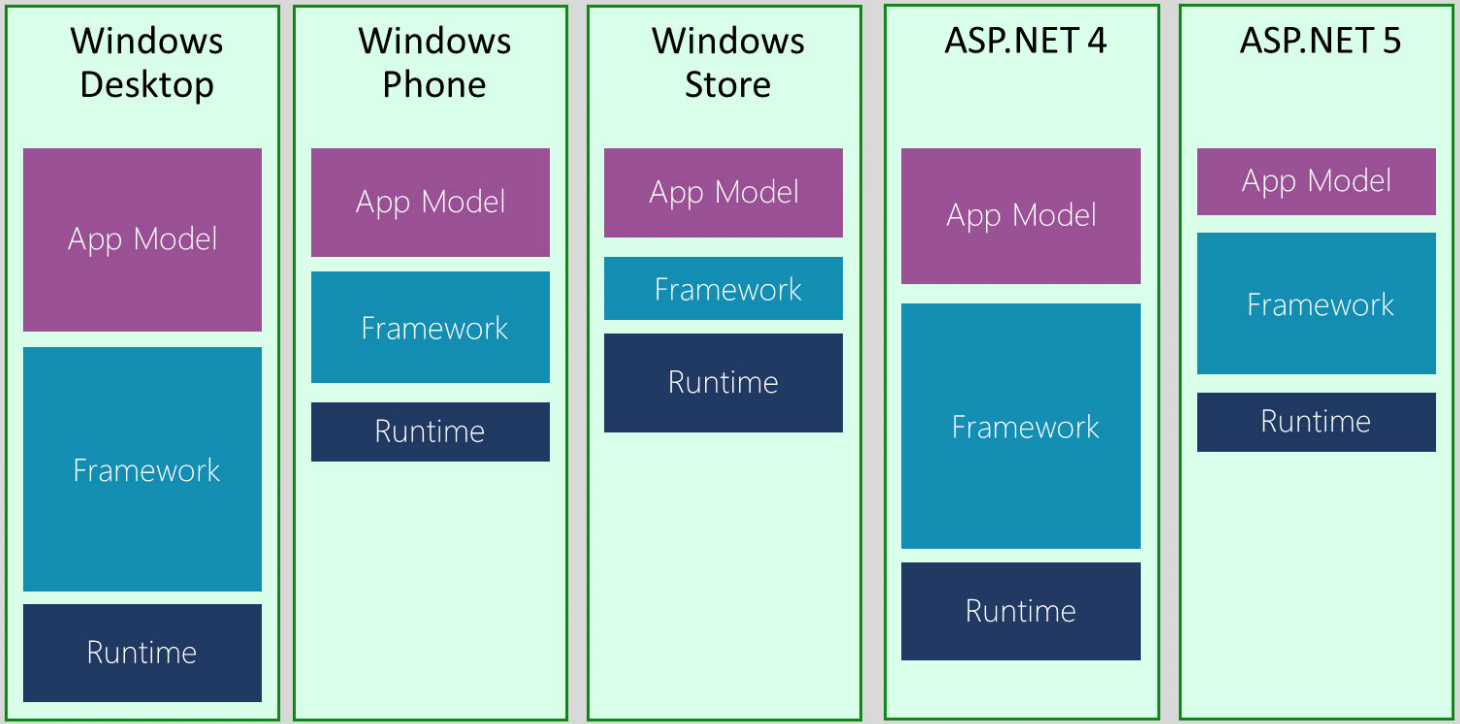
Overview

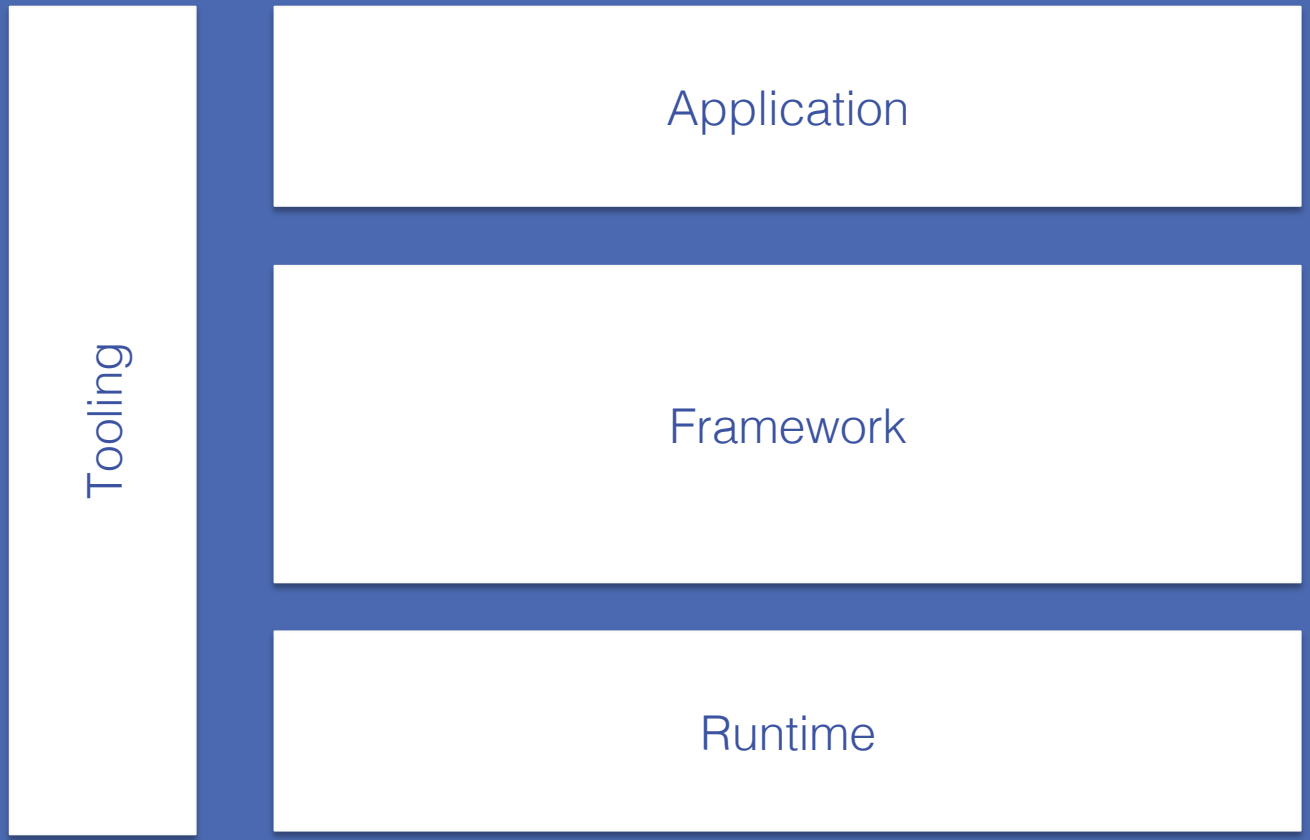


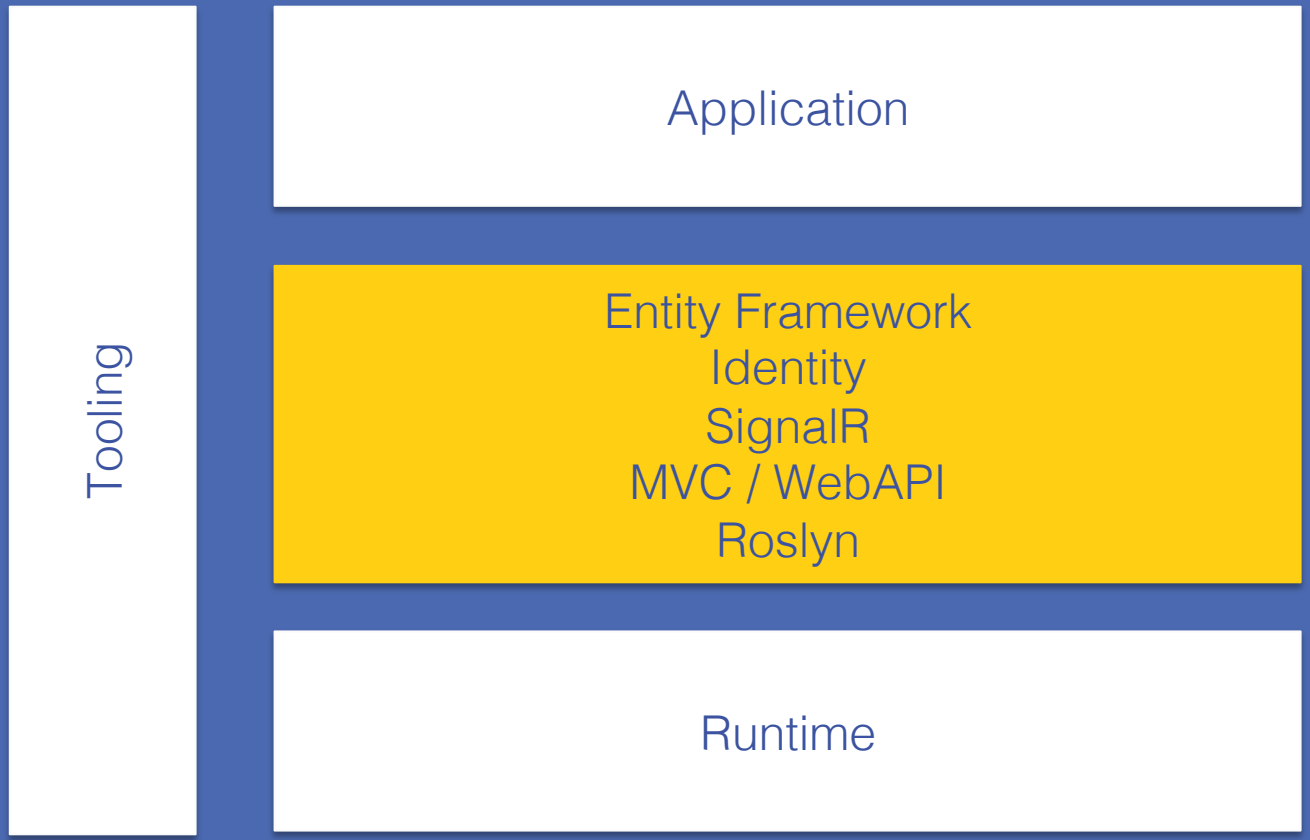
What is .NET?

- Introduced in 2002
- It provides
  - Core class libraries
  - Complier
  - Runtime (execution layer)
  - Support for CLI languages (C#, VB, plus more)

.NET


Windows Desktop
App Model
Framework
Runtime
Windows Phone
App Model
Framework
Runtime
Windows Store
App Model
Framework
Runtime
ASP.NET 4
App Model
Framework
Runtime
ASP.NET 5
App Model
Framework
Runtime


Application
Framework
Runtime
Tooling


Tooling
Application
Entity Framework
Identity
SignalR
MVC / WebAPI
Roslyn
Runtime

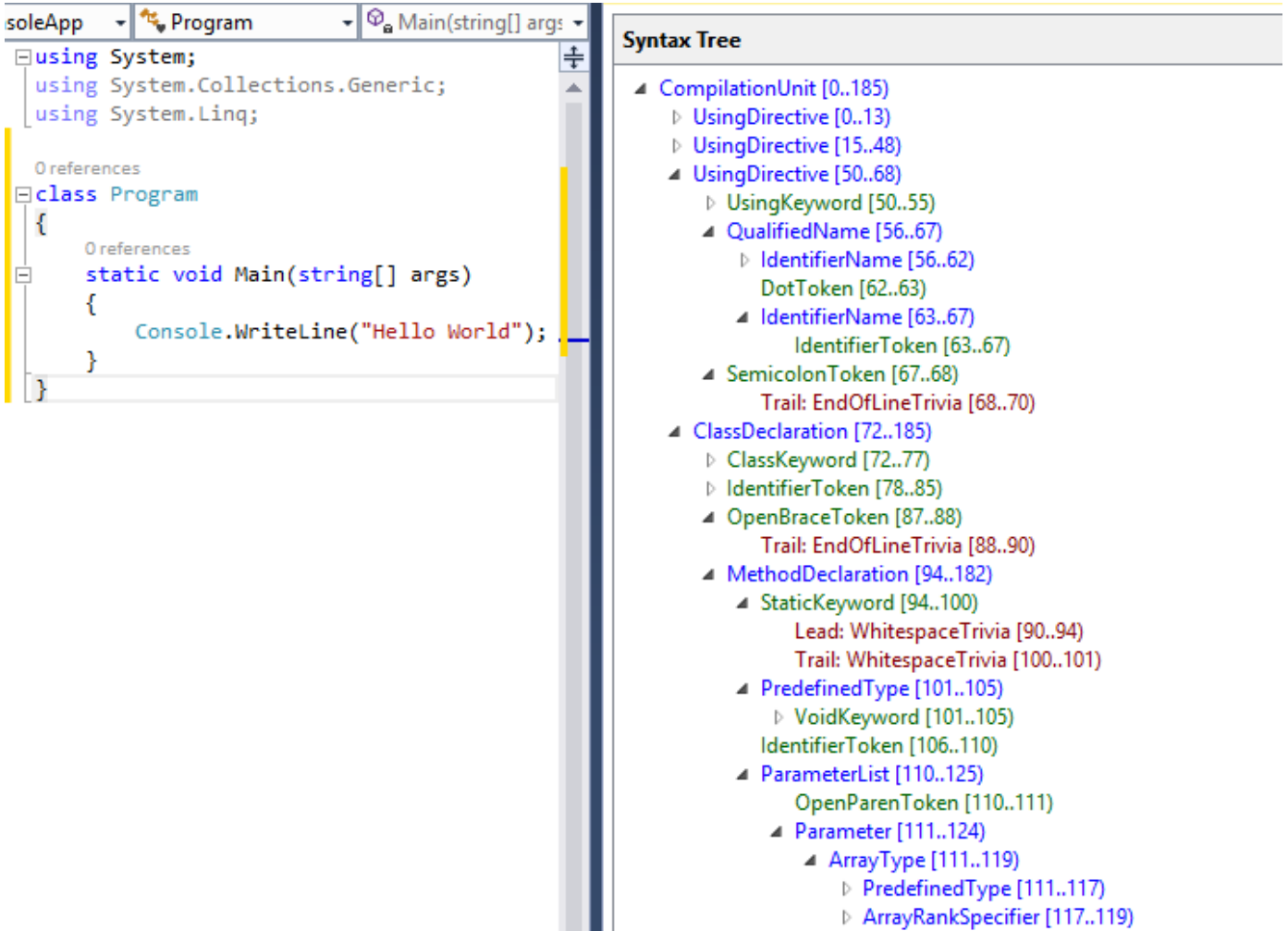
Frameworks

Roslyn

What is Roslyn?

- Code Analysis APIs

- .NET Core


soleApp
Program
Main(string[] args
using System;
using System.Collections.Generic;
using System.Linq;
0 references
class Program
{
0 references
static void Main(string[] args)
{
Console.WriteLine("Hello World");
}
}
Syntax Tree
CompilationUnit [0..185)
UsingDirective [0..13)
UsingDirective [15..48)
UsingDirective [50..68)
UsingKeyword [50..55)
QualifiedName [56..67)
IdentifierName [56..62)
DotToken [62..63)
IdentifierName [63..67)
IdentifierToken [63..67)
SemicolonToken [67..68)
Trail: EndOfLineTrivia [68..70)
ClassDeclaration [72..185)
ClassKeyword [72..77)
IdentifierToken [78..85)
OpenBraceToken [87..88)
Trail: EndOfLineTrivia [88..90)
MethodDeclaration [94..182)
StaticKeyword [94..100)
Lead: WhitespaceTrivia [90..94)
Trail: WhitespaceTrivia [100..101)
PredefinedType [101..105)
VoidKeyword [101..105)
IdentifierToken [106..110)
ParameterList [110..125)
OpenParenToken [110..111)
Parameter [111..124)
ArrayType [111..119)
PredefinedType [111..117)
ArrayRankSpecifier [117..119)

What is Roslyn?

- Code Analysis APIs
- .NET Core

- “Introduction to Roslyn” at 2:40pm today

Entity Framework

Entity Framework

- Data access framework

```
db.Animals.Where(i=>i.Name == “penguin”).Take(4)

SELECT * FROM animals WHERE name=“penguin” LIMIT 4
```

Entity Framework 6

- MSSQL and MySQL
- Primarily ASP.NET

What's New in EF 7

- Azure Table Storage, Redis
- SQLite

- Complete rewrite
- New APIs (similar to v6)
- CLI tools

What’s new in v7

- Cloud optimized
- Runs on .NET Core

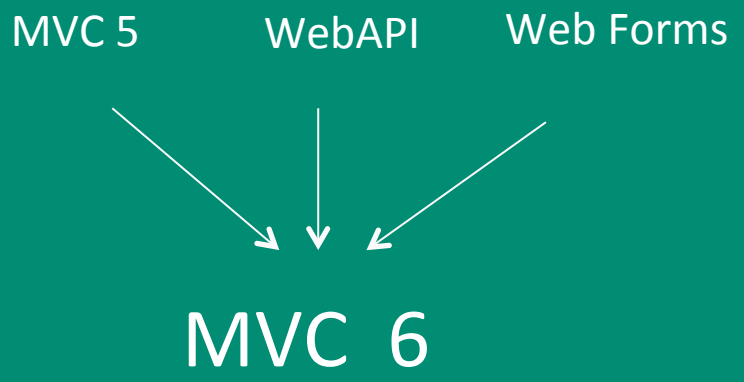
MVC

MVC

- The .NET web framework

What's new in v6

express



- Host agnostic
- Built-in dependency injection
- config.json

What’s new in v6

- Cloud optimized
- Runs on .NET Core

Identity

Identity

- User authentication framework
- OAuth providers and two-factor auth

What’s new in v6

- Cloud optimized
- Runs on .NET Core

SignalR

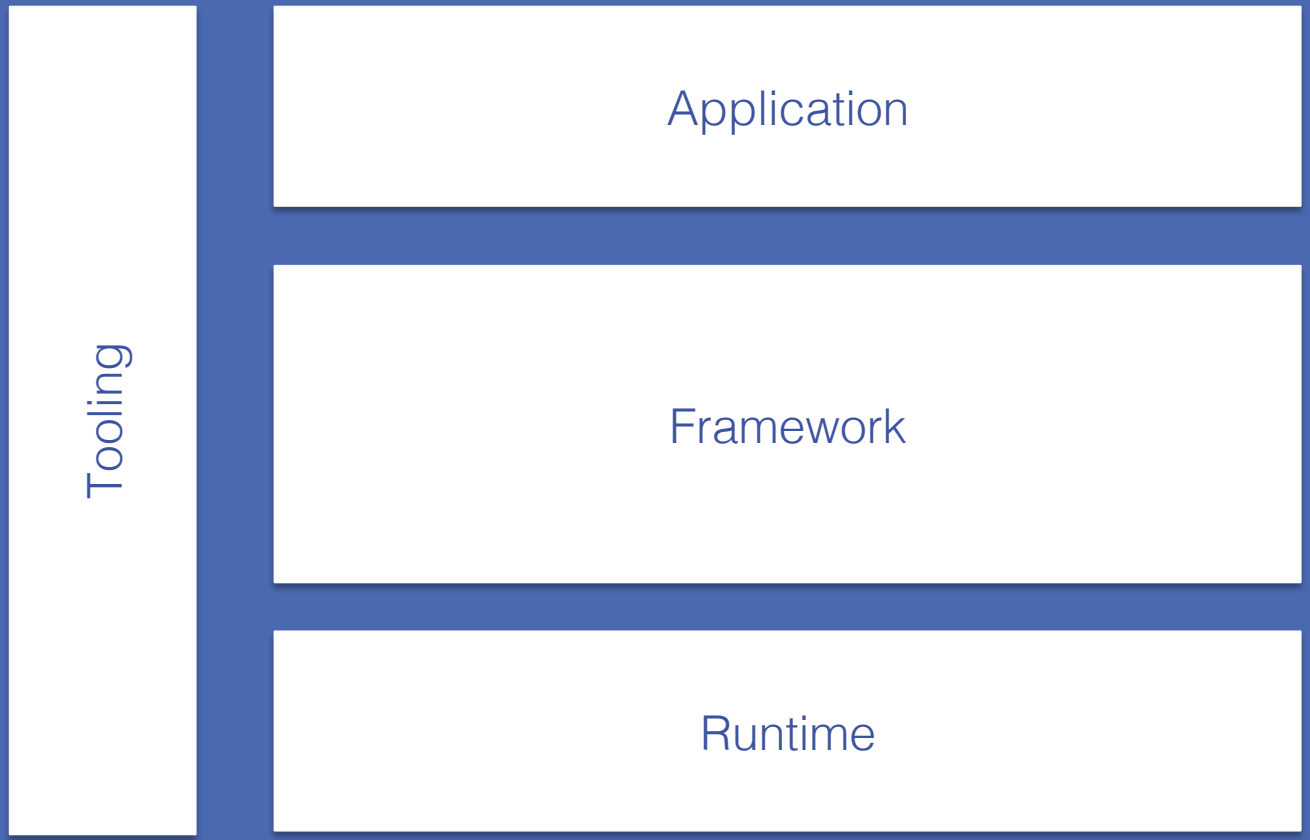
SignalR

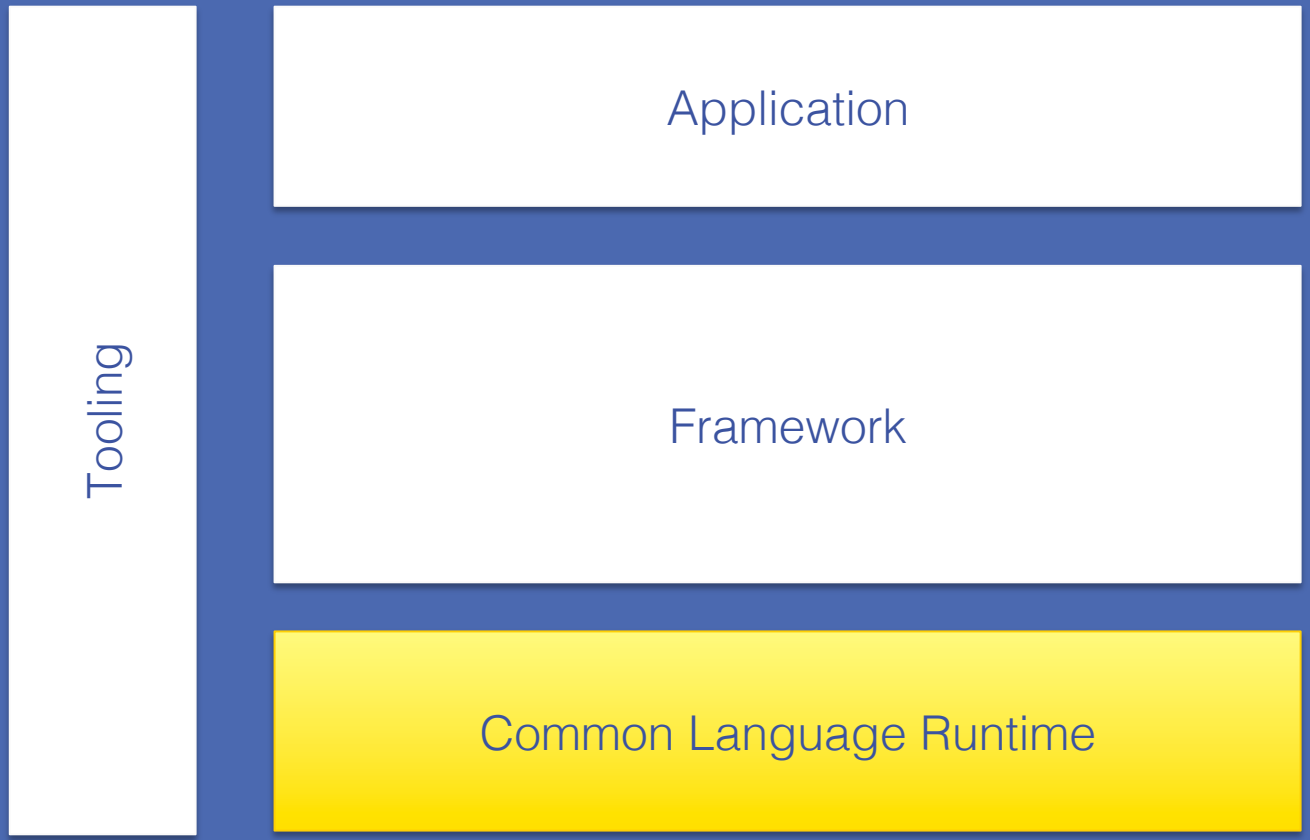
- Real-time web functionality
- Server and client components

Not yet

- Not cloud optimized
- Does not run on .NET Core

What is .NET Core?
What is “cloud optimized”?

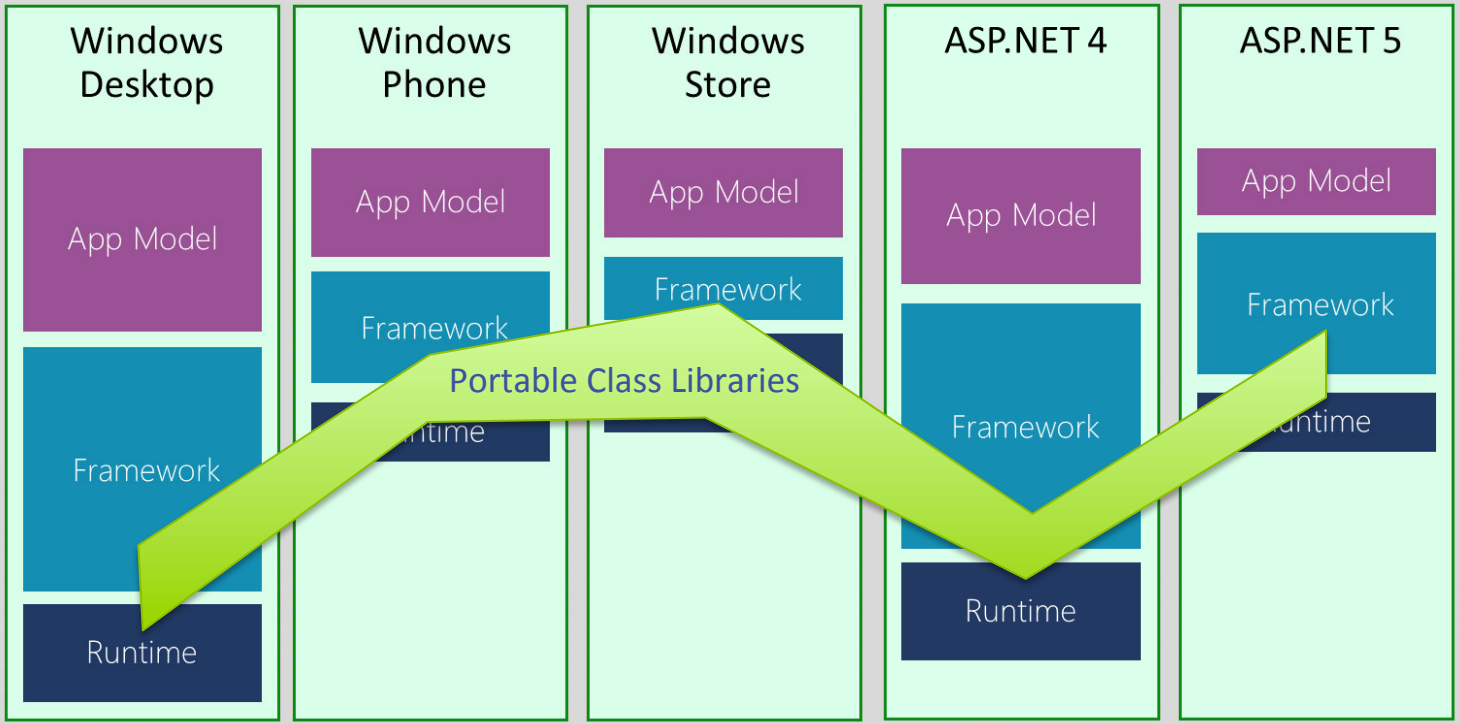

Application
Framework
Runtime
Tooling


Tooling
Application
Framework
Common Language Runtime

.NET 4.6

- Improve cross-platform support
- New cryptography APIs
- Task-based asynchronous pattern (TAP)
- Additional data collections Stack<T>
  ReadOnlyCollection

.NET

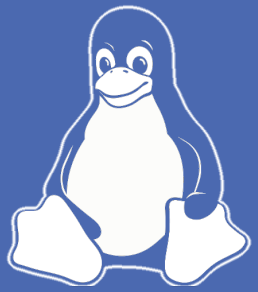


+ Silverlight, Xbox apps, and legacy .NET

Uncommon Language Runtime?

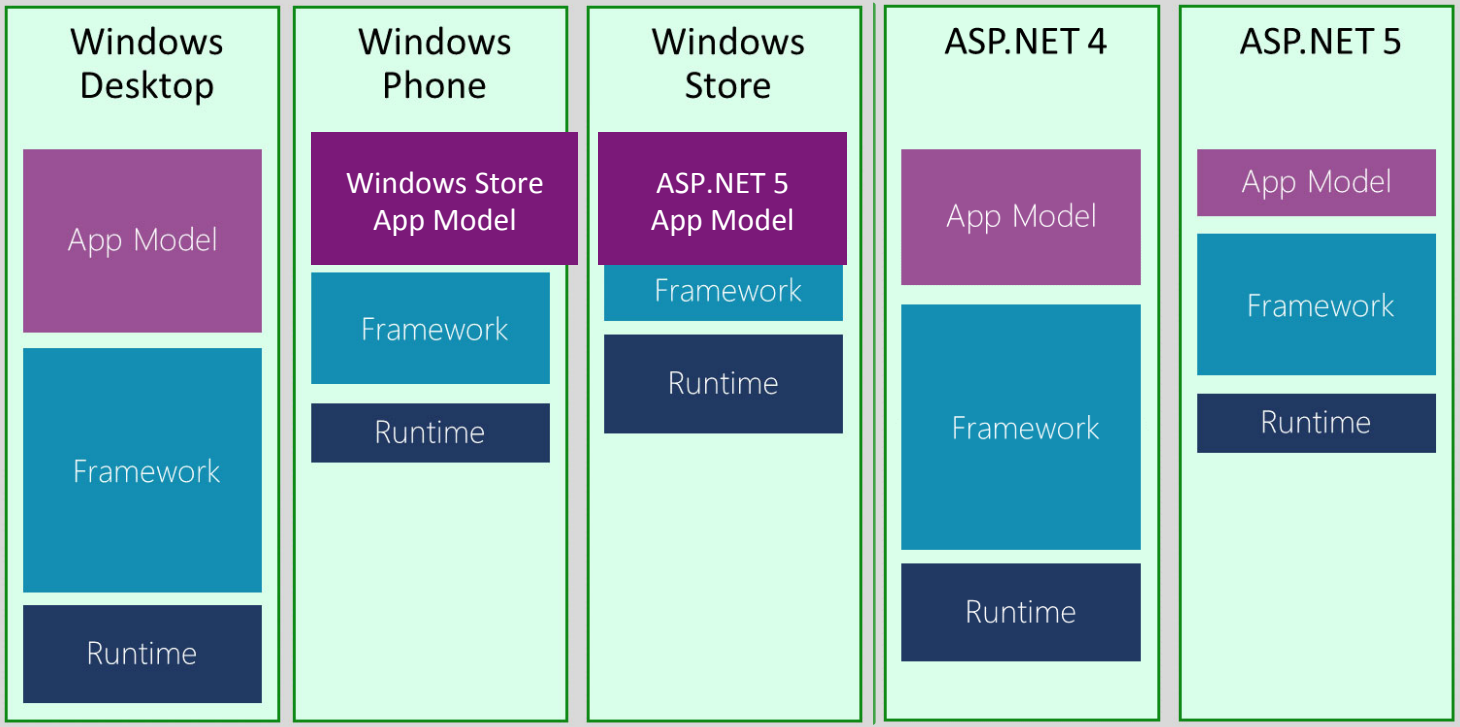
So fork it…

.NET Core

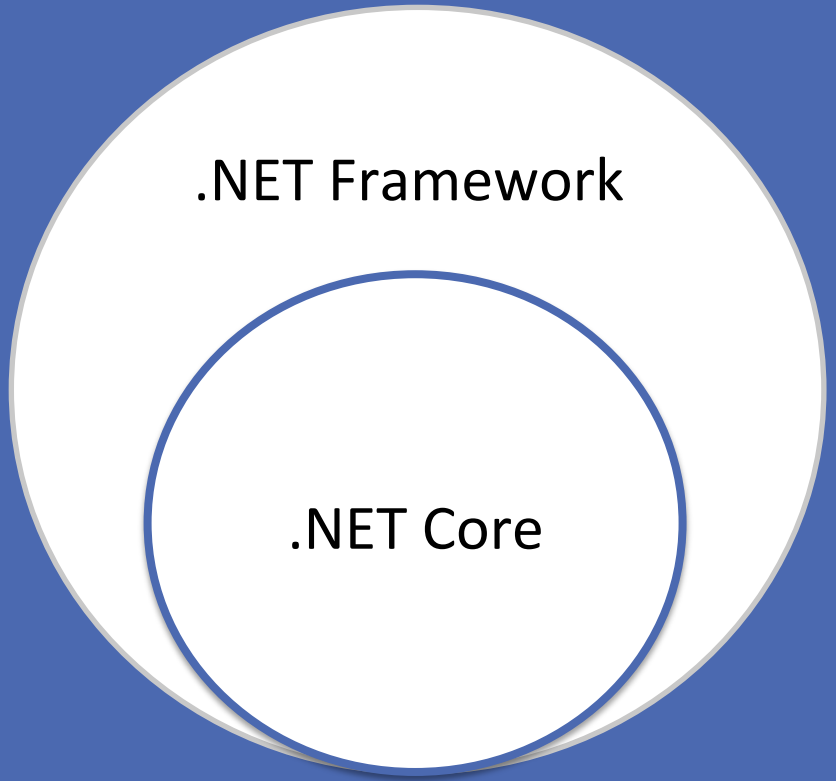


GitHub

.NET


Windows Desktop
App Model
Framework
Runtime
Windows Phone
Windows Store App Model
Framework
Runtime
Windows Store
ASP.NET 5 App Model
Framework
Runtime
ASP.NET 4
App Model
Framework
Runtime
ASP.NET 5
App Model
Framework
Runtime

API Surface



.NET 2015

.NET Framework

ASP.NET 5
ASP.NET 4.6
WPF
Windows Forms

.NET Core

ASP.NET 5
.NET Native

ASP.NET 5 for Mac and Linux

Common

Runtime

Next gen JIT
SIMD

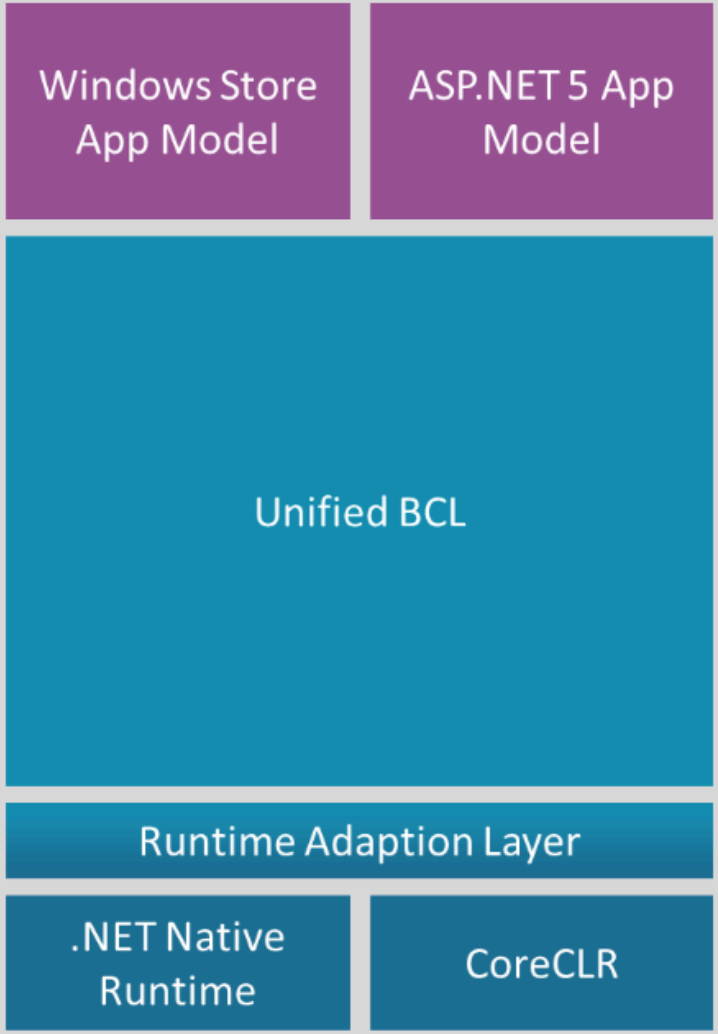
Compilers

.NET Compiler Platform
Languages innovation

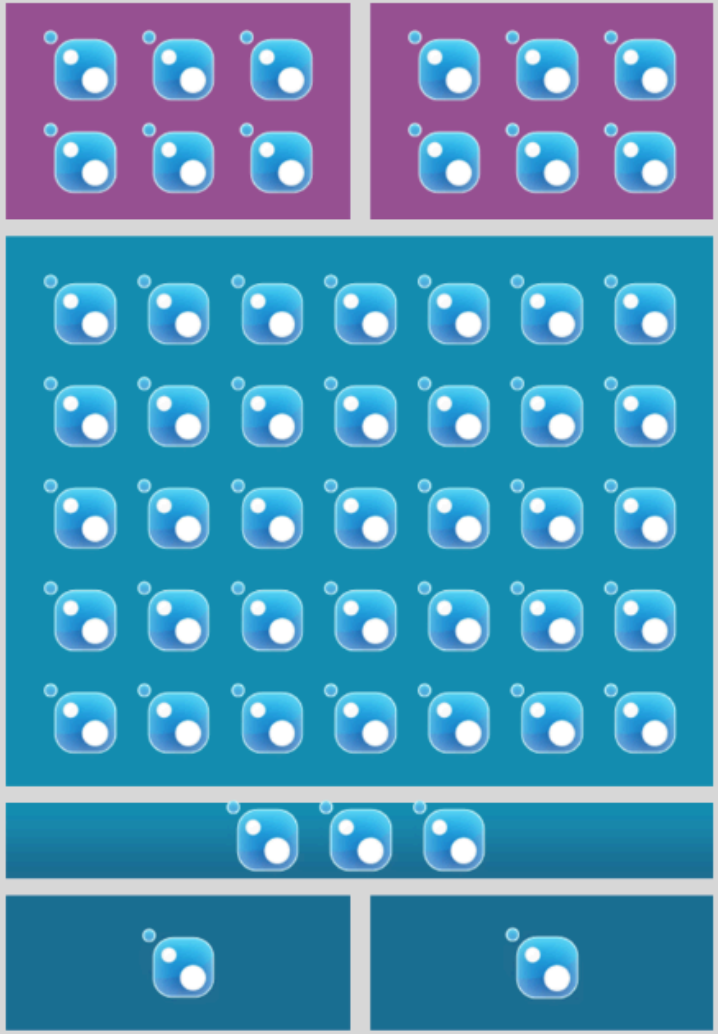
NuGet packages

.NET Core 5 Libraries
.NET Framework 4.6 Libraries

.NET Core


Windows Store App Model
ASP.NET 5 App Model
Unified BCL
Runtime Adaption Layer
.NET Native Runtime
CoreCLR

.NET Core

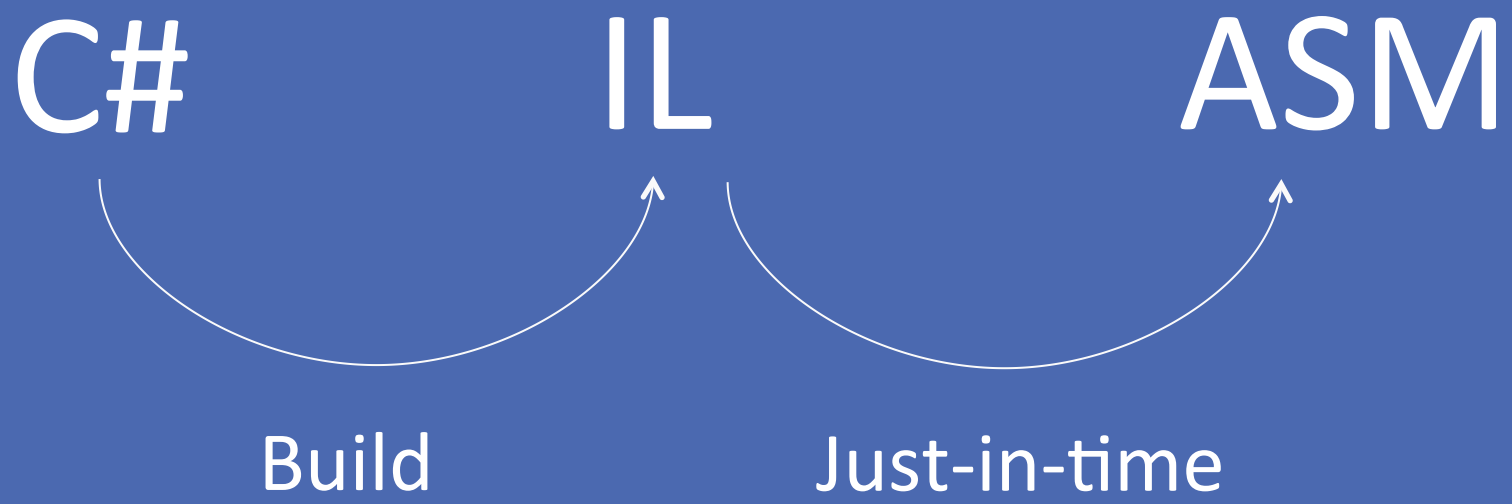


Cloud Optimized

- Distributed in packages
- App-local
- Optimized for memory

.NET Native

CLR and IL



Demo

Tooling

CLI Tools

- Beta 3
  - KPM, KVM, KLR, K
- Beta 4
  - DNVM, DNX

Visual Studio 2015

What's New

- Modern web tooling
  - Javascript Debugger
  - Improved editor
- Refactoring tools (Roslyn)
- .NET Core and ASP.NET 5
- Git
- Android and iOS developer tools (Xamarin)
- Android emulator

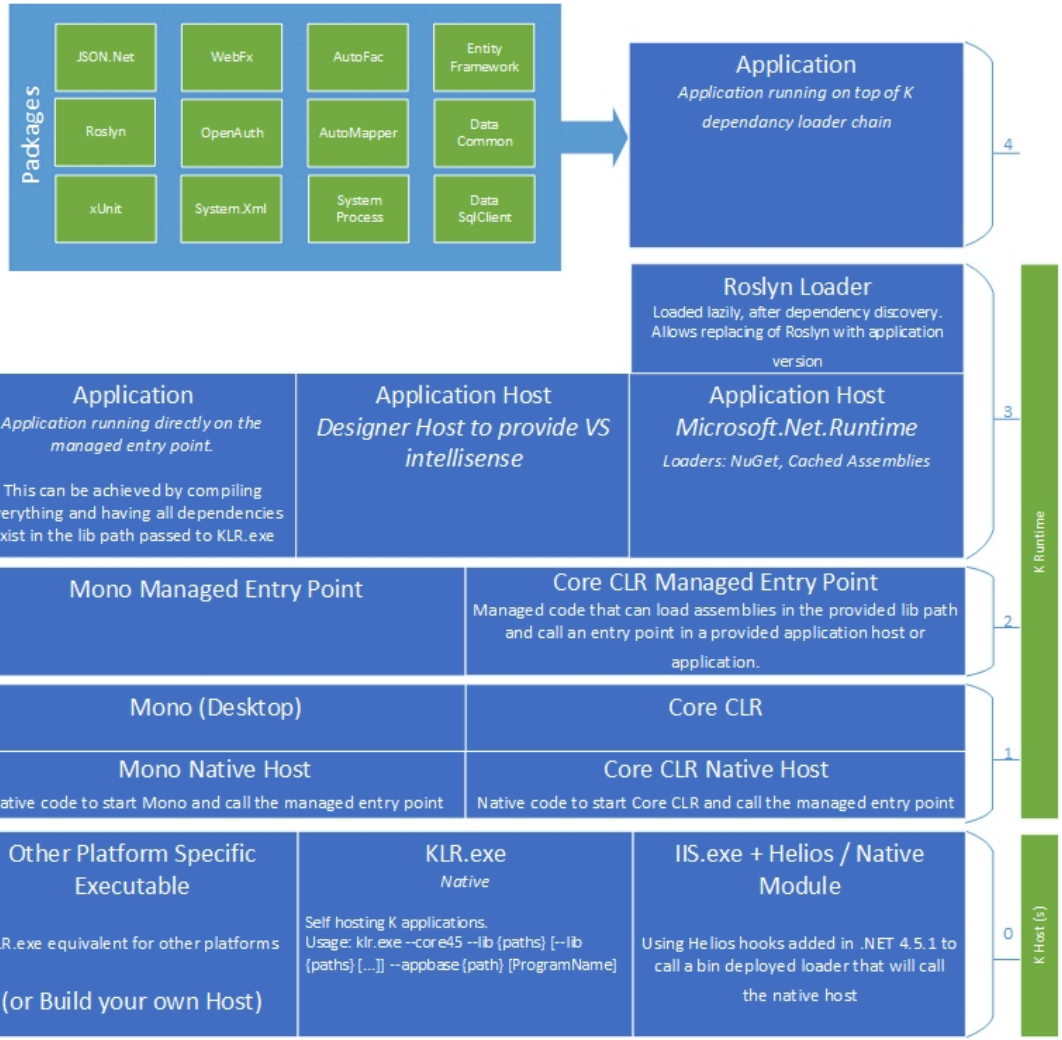
Modernizing

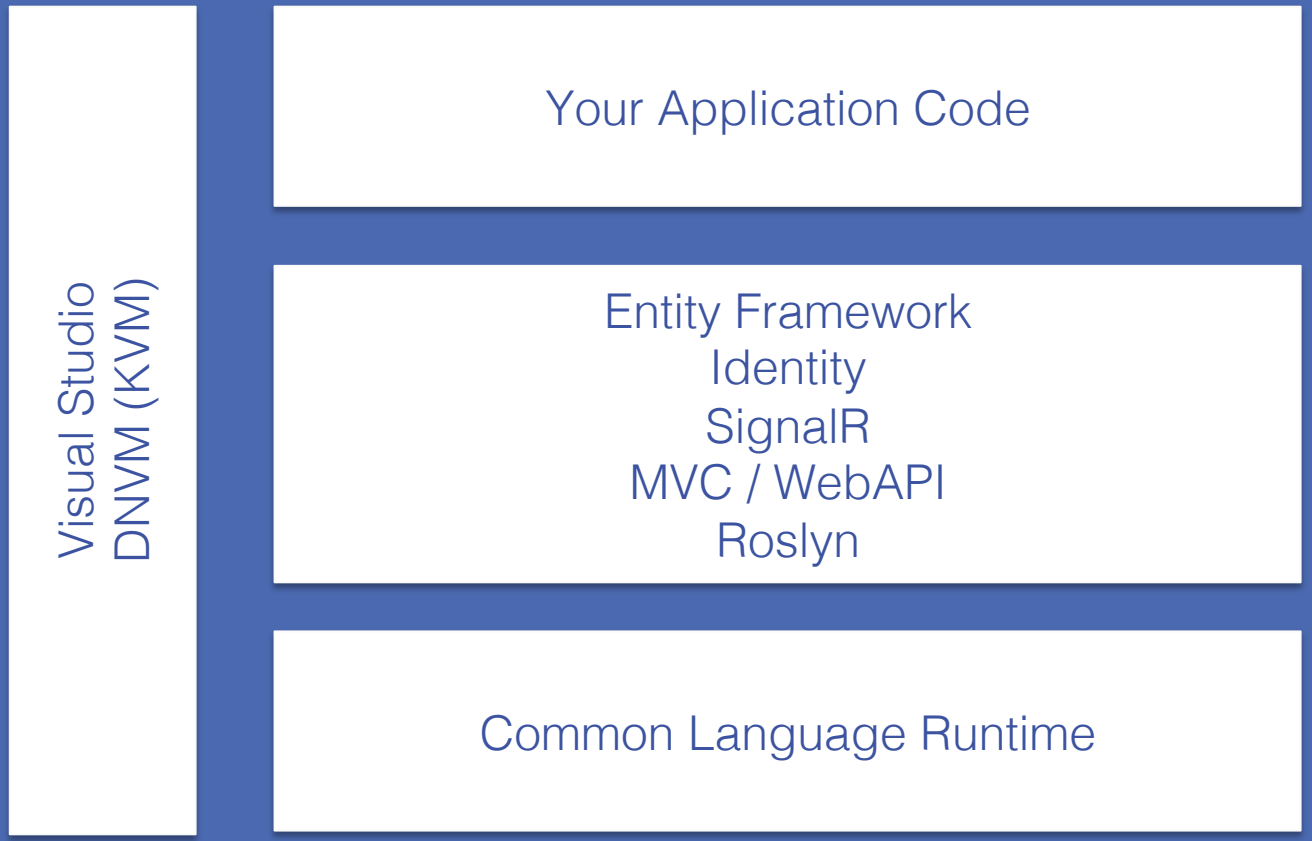
- Web tools
  - Grunt, gulp, bower, NPM
- Mobile development tools
  - Cordova debugging
  - Android emulator
  - Responsive web design

Nate McMaster
Twitter @natemcmaster
GitHub github.com/natemcmaster

Thanks to our Sponsors!

To connect to wireless
1. Choose Uguest in the wireless list
2. Open a browser. This will open a Uof U website
3. Choose Login


Packages
JSON.Net
WebFx
AutoFac
Entity Framework
Roslyn
OpenAuth
AutoMapper
Data Common
xUnit
System.Xml
System Process
Data SqlClient
Application
Application running on top of K dependancy loader chain
4
Roslyn Loader
Loaded lazily, after dependency discovery. Allows replacing of Roslyn with application version
Application
Application running directly on the managed entry point.
This can be achieved by compiling everything and having all dependencies exist in the lib path passed to KLR.exe
Application Host
Designer Host to provide VS intellisense
Application Host
Microsoft.Net.Runtime
Loaders: NuGet, Cached Assemblies
3
Mono Managed Entry Point
Core CLR Managed Entry Point
Managed code that can load assemblies in the provided lib path and call an entry point in a provided application host or application.
2
Mono (Desktop)
Core CLR
Mono Native Host
Native code to start Mono and call the managed entry point
Core CLR Native Host
Native code to start Core CLR and call the managed entry point
1
K Runtime
Other Platform Specific Executable
KLR.exe equivalent for other platforms
(or Build your own Host)
KLR.exe
Native
Self hosting K applications.
Usage: klr.exe --core45 --lib {paths} [--lib {paths} [...]] --appbase {path} [ProgramName]
IIS.exe + Helios / Native Module
Using Helios hooks added in .NET 4.5.1 to call a bin deployed loader that will call the native host
0
K Host (s)


Visual Studio
DNVM (KVM)
Your Application Code
Entity Framework
Identity
SignalR
MVC / WebAPI
Roslyn
Common Language Runtime