.NET Core

infogain

Objective
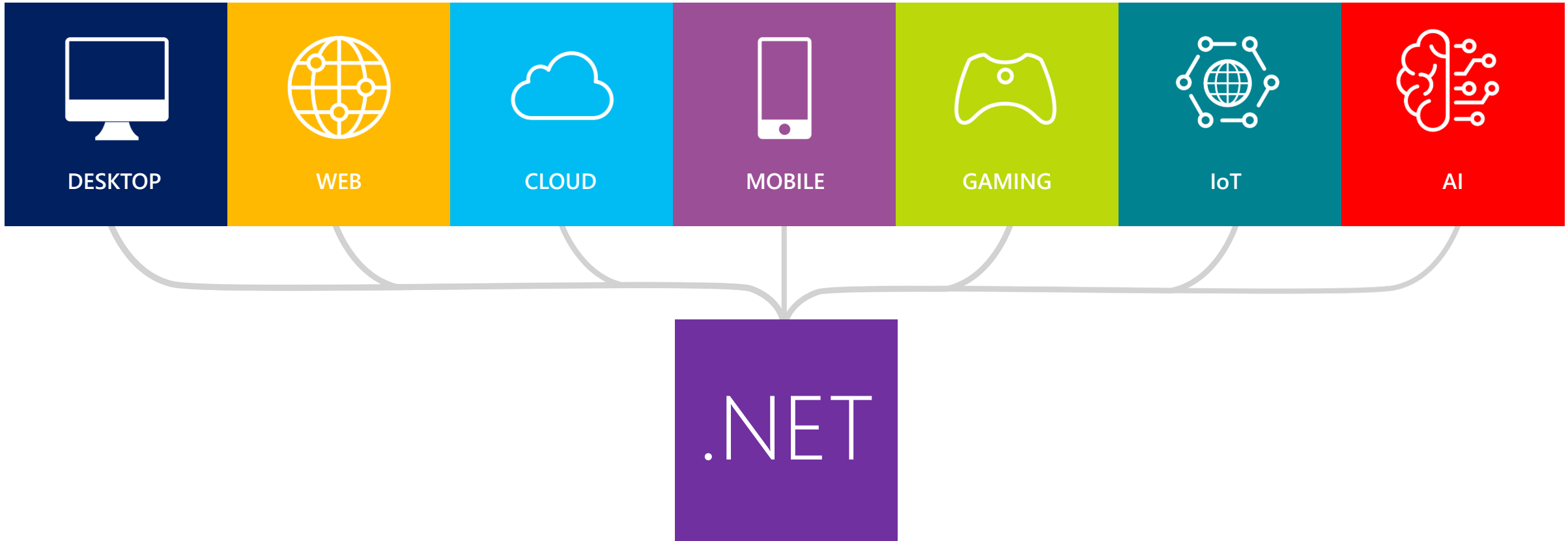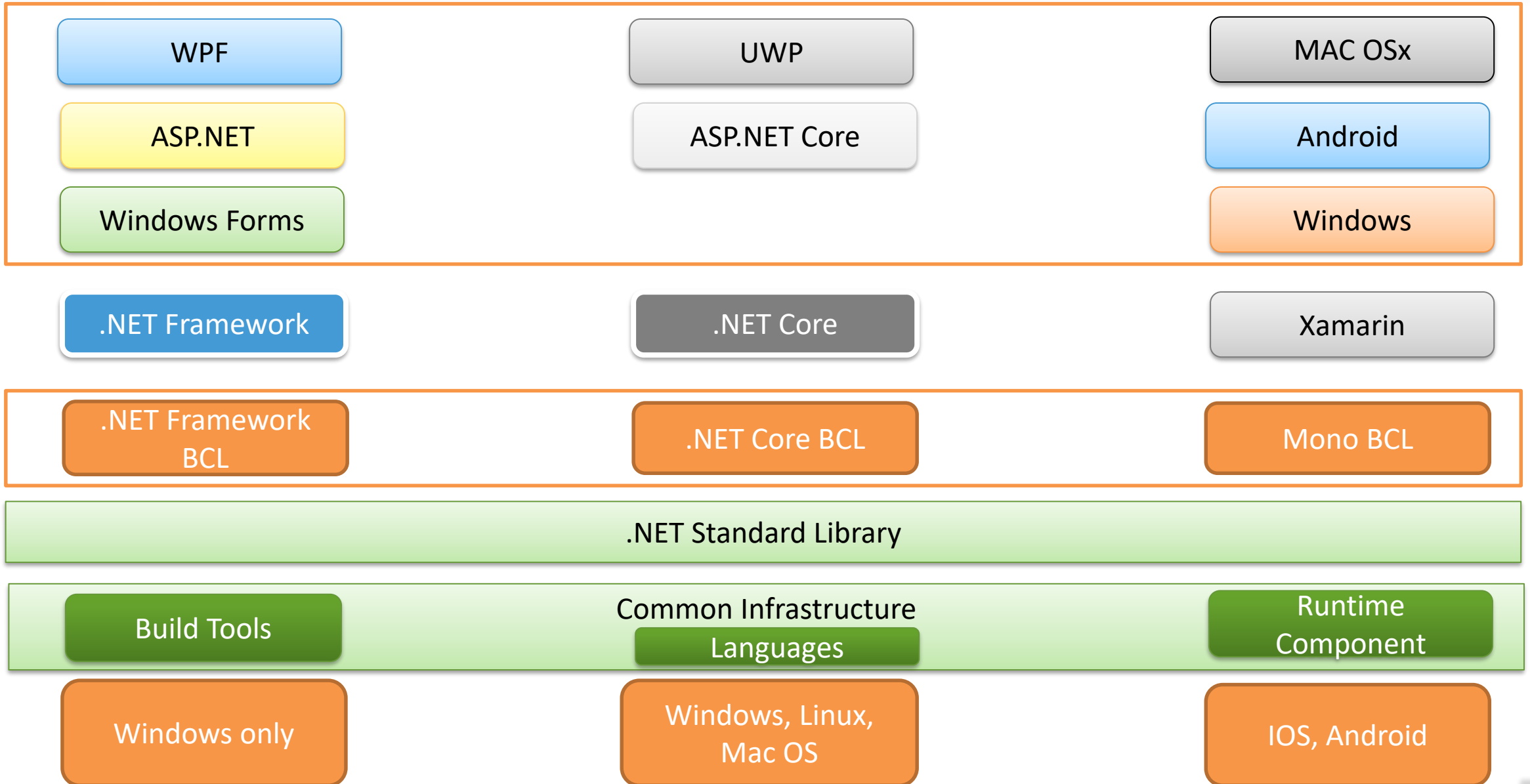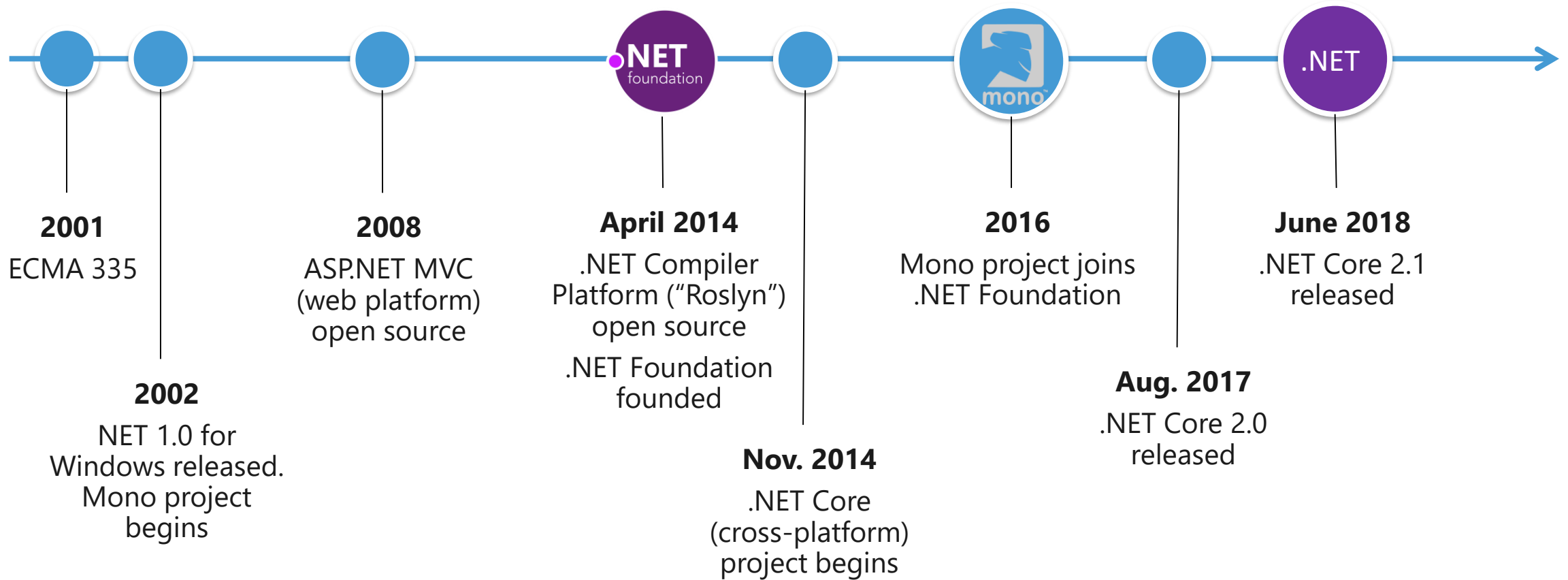
- Understanding the .NET Ecosystem
- Comparison of runtime .NET Core vs. .NET framework vs. Xamarin
- .NET Standard specification
- .NET Core journey
- .NET Core App Demo
- Different version of .NET Core
- Asp.Net (.NET framework) vs Asp.Net Core

Your platform for building **anything**

| DESKTOP | WEB | CLOUD | MOBILE | GAMING | IoT | AI |

.NET

Confidential For Internal Discussion Purposes Only

Needs for the runtime

- Cross platform deployment
- Open source
- Give customer choice
  - Application local
  - Side by side installations
- Innovate fast and ship fast

Confidential For Internal Discussion Purposes Only

.NET Ecosystem

WPF	UWP	MAC OSx
ASP.NET	ASP.NET Core	Android
Windows Forms		Windows

.NET Framework	.NET Core	Xamarin

.NET Framework BCL	.NET Core BCL	Mono BCL

.NET Standard Library

Common Infrastructure

Build Tools	Languages	Runtime Component

Windows only	Windows, Linux, Mac OS	IOS, Android

Confidential For Internal Discussion Purposes Only

Confidential For Internal Discussion Purposes Only

.NET Open Source Journey

2001

ECMA 335

2002

NET 1.0 for Windows released. Mono project begins

2008

ASP.NET MVC (web platform) open source

April 2014

.NET Compiler Platform ("Roslyn") open source

.NET Foundation founded

Nov. 2014

.NET Core (cross-platform) project begins

2016

Mono project joins .NET Foundation

Aug. 2017

.NET Core 2.0 released

June 2018

.NET Core 2.1 released

Learning

- C# compiler goes "OSS" on CodePlex

- Source Open not Open Source
  - Roslyn team still worked primarily in TFS
  - Only 1/3 of the code base was available
  - Regularly pushed updates to CodePlex

- Results were mixed
  - Did receive a number of PRs
  - No sustained contribution

Confidential For Internal Discussion Purposes Only

Move to GitHub

- This is where our customers were

- Embraced living on GitHub
  - No more working in closed and pushing to open
  - All development done on GitHub
  - Work how you want your contributors to work

- Learn the Github way
  - How to compose a product
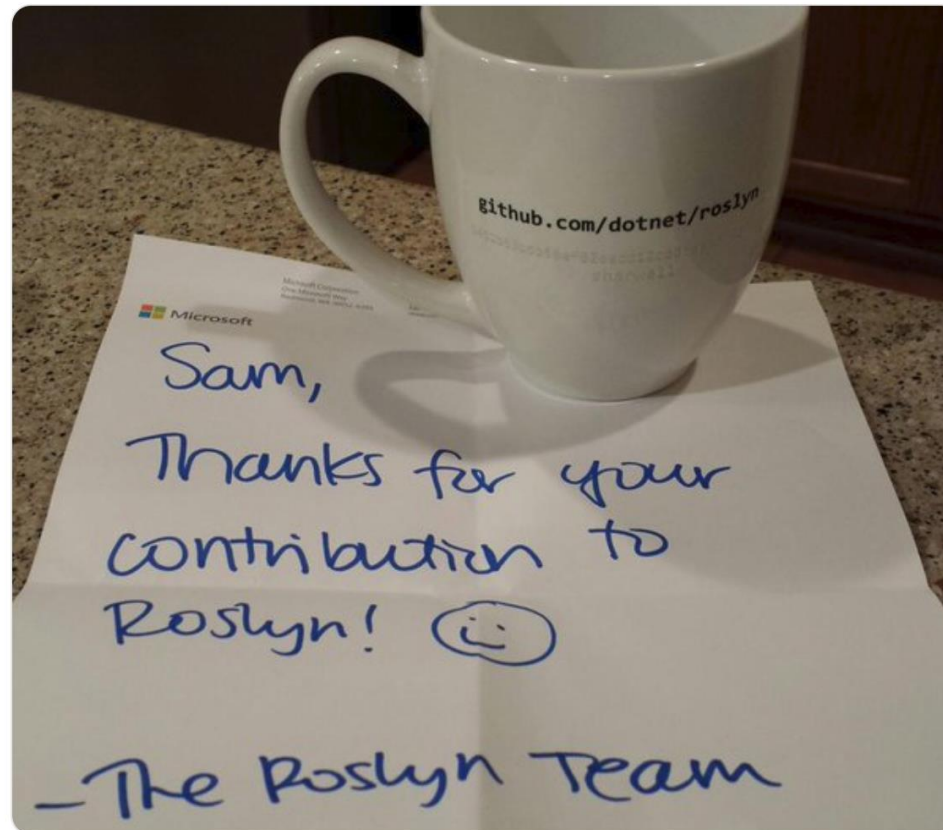 - Developer work flow
  - How to ship/service a product

Confidential For Internal Discussion Purposes Only
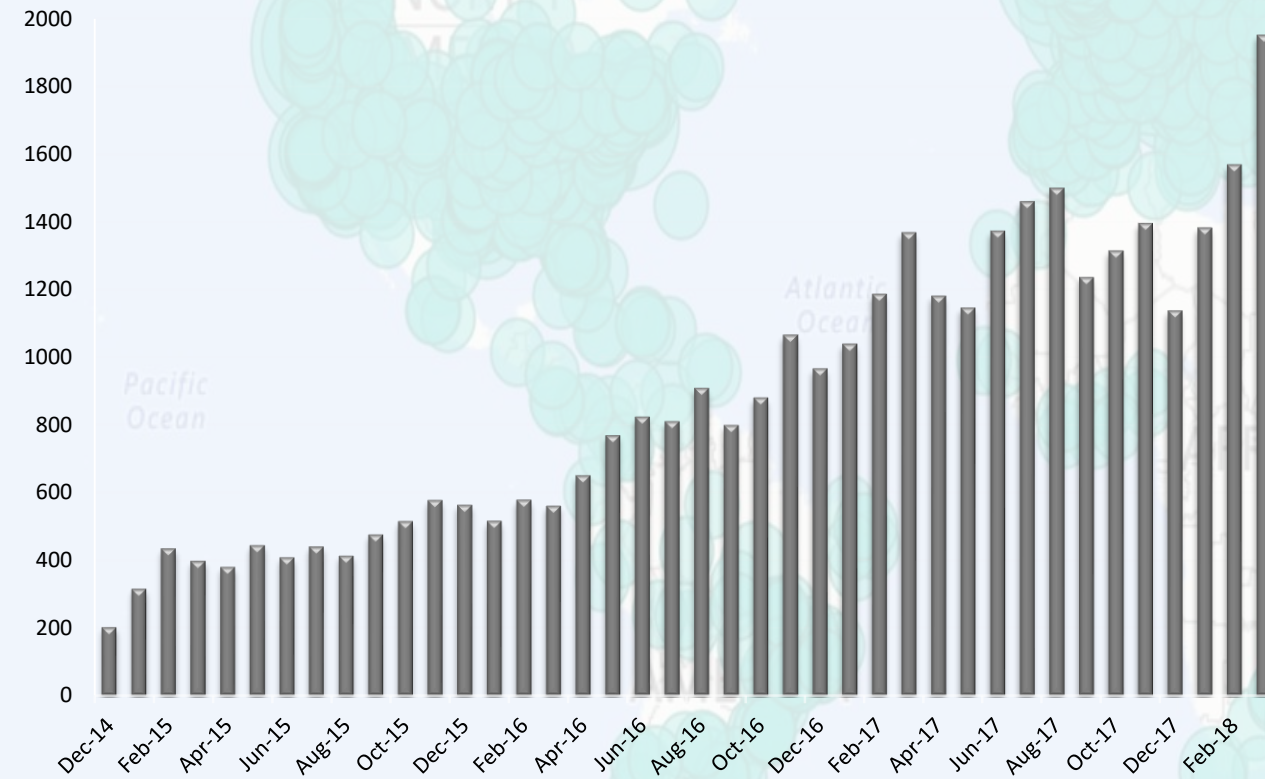
Thanking the community

.NET Open Source Success
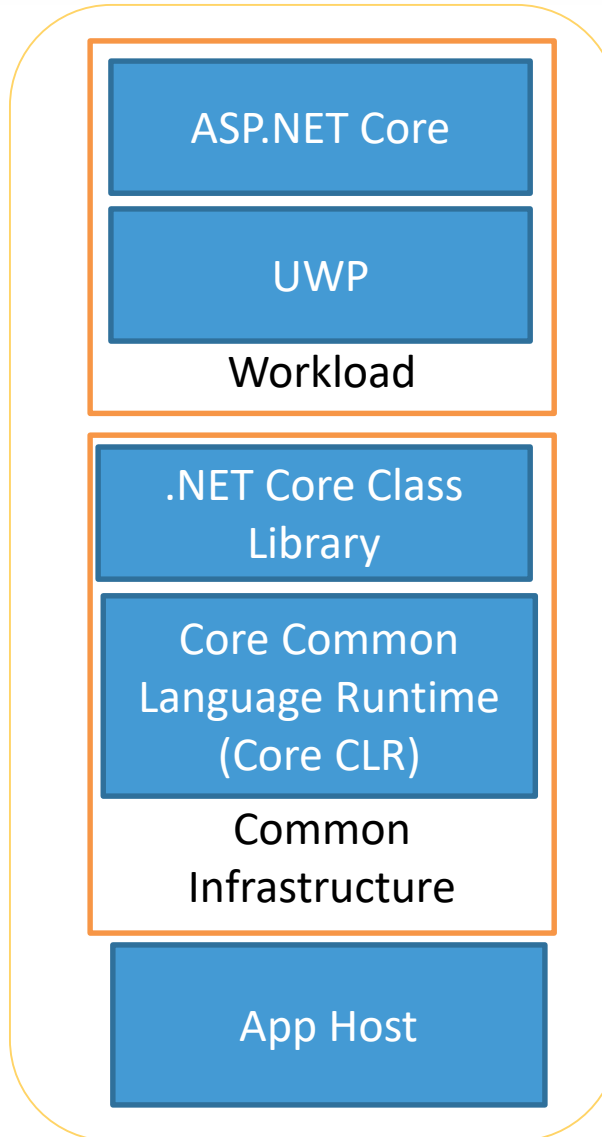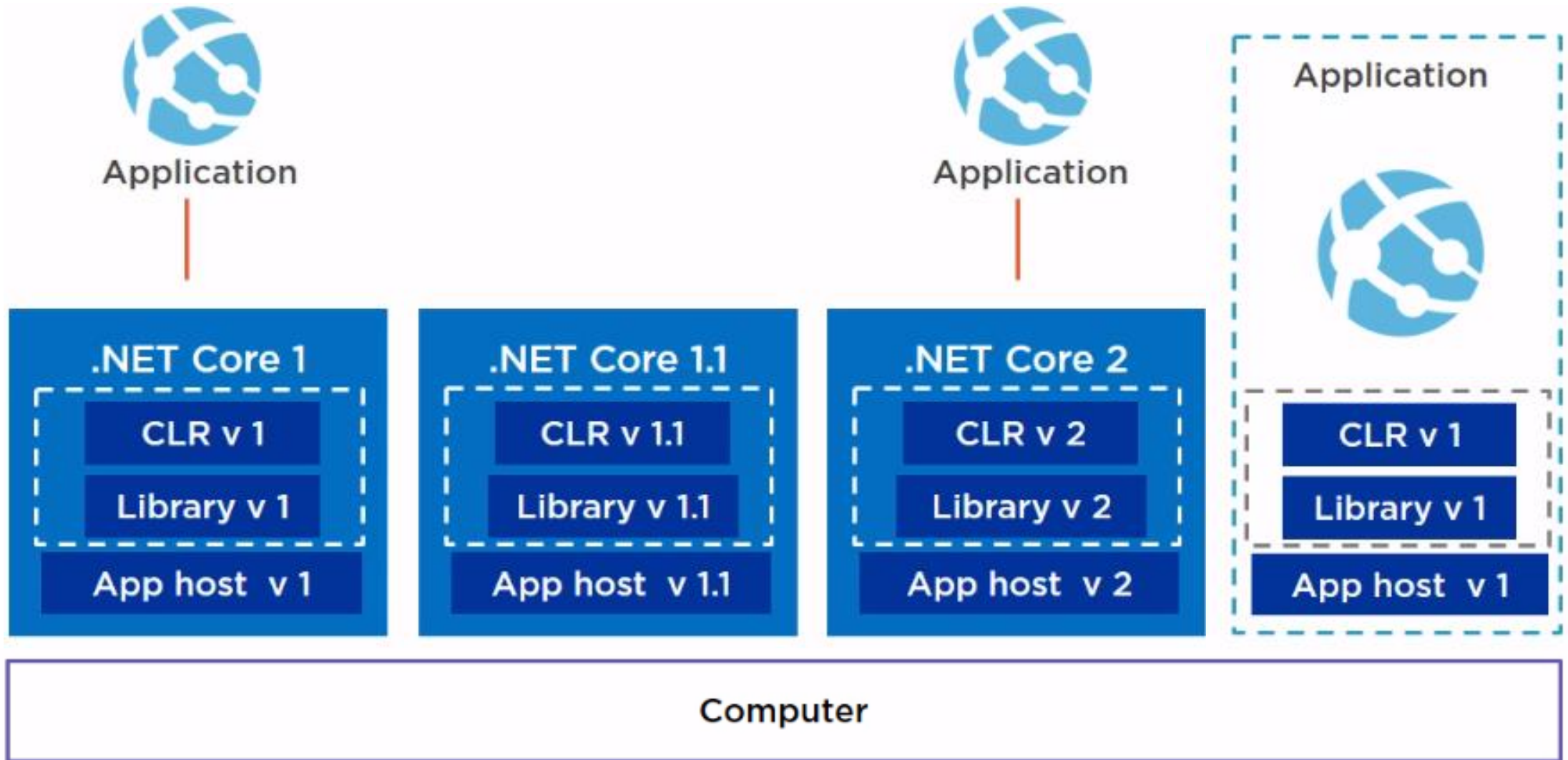
COMMUNITY ACCEPTED PULL REQUESTS

"Samsung is embracing .NET because it is a completely open source project." — Samsung

".NET is open source; that allows us to contribute back to it if we have any performance issues which Microsoft review and together we make a better product." — Illyriad Games

20,000+ Community Contributions from 3700+ Companies outside Microsoft
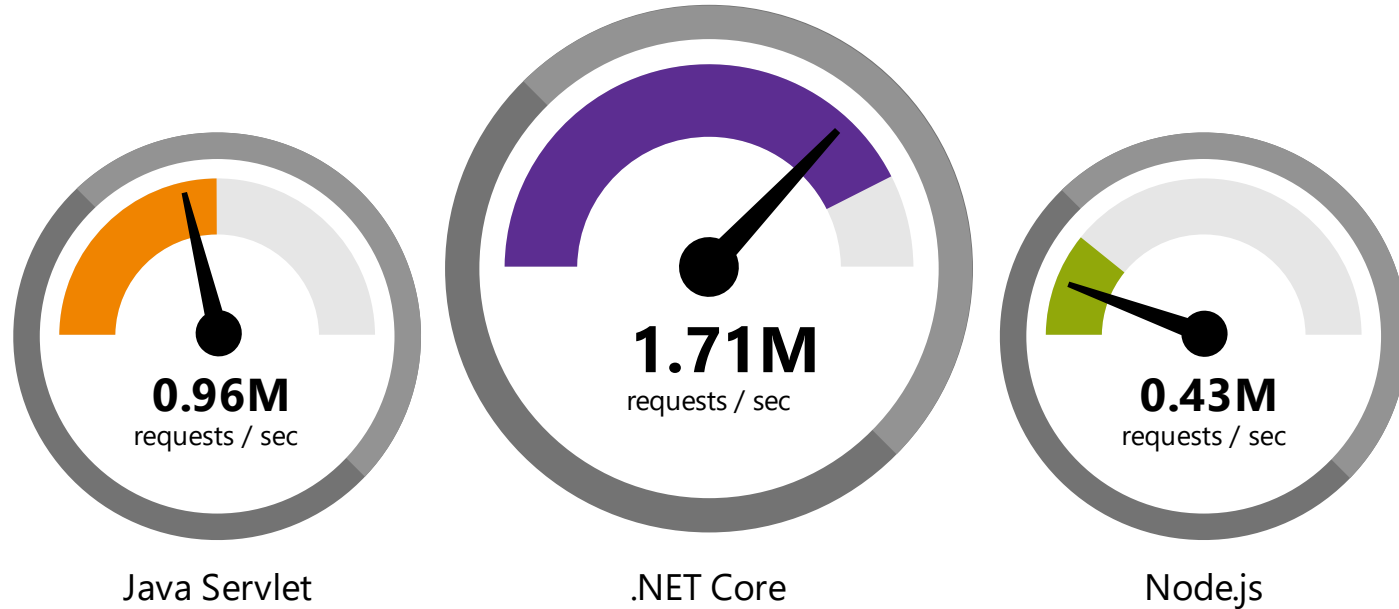
.NET Core 1.X

- Released in 2016
- Deliver a cross platform runtime: Windows, Linux and OSX
- Slim down our runtime/framework
- Flexible deployment
- Open source
    - https://github.com/dotnet/core

Confidential For Internal Discussion Purposes Only

.NET Core

Workload
ASP.NET Core
UWP

Common Infrastructure
.NET Core Class Library
Core Common Language Runtime (Core CLR)

App Host

Confidential For Internal Discussion Purposes Only

.NET Core

.NET Core 1	.NET Core 1.1	.NET Core 2	Application
CLR v 1	CLR v 1.1	CLR v 2	CLR v 1
Library v 1	Library v 1.1	Library v 2	Library v 1
App host v 1	App host v 1.1	App host v 2	App host v 1

Computer

Confidential For Internal Discussion Purposes Only   14

Lessons Learned

- Only new code can target 1.X
  - Trimmed down the API set too far
  - Porting existing applications extremely difficult
  - netstandard insufficient for writing portable applications

- We went too far away from existing customers

- .NET Standard 1.X not ideal for portable applications

.NET Core 2.X

- Performance
- Expand the API set
- Developer tooling experience

Confidential For Internal Discussion Purposes Only

.NET Core 2 is Fast

infogain

Java Servlet	.NET Core	Node.js
0.96M requests / sec	**1.71M** requests / sec	**0.43M** requests / sec

Data sourced from official tests available at TechEmpower Round 14.

"Using the same-size server, we were able to go from 1,000 requests per second per node with Node.js to 20,000 requests per second with .NET Core."
— Raygun

https://www.microsoft.com/net/customers

.NET Core 2.1 is Faster!

RUNTIME PERFORMANCE

Expanding the API Set
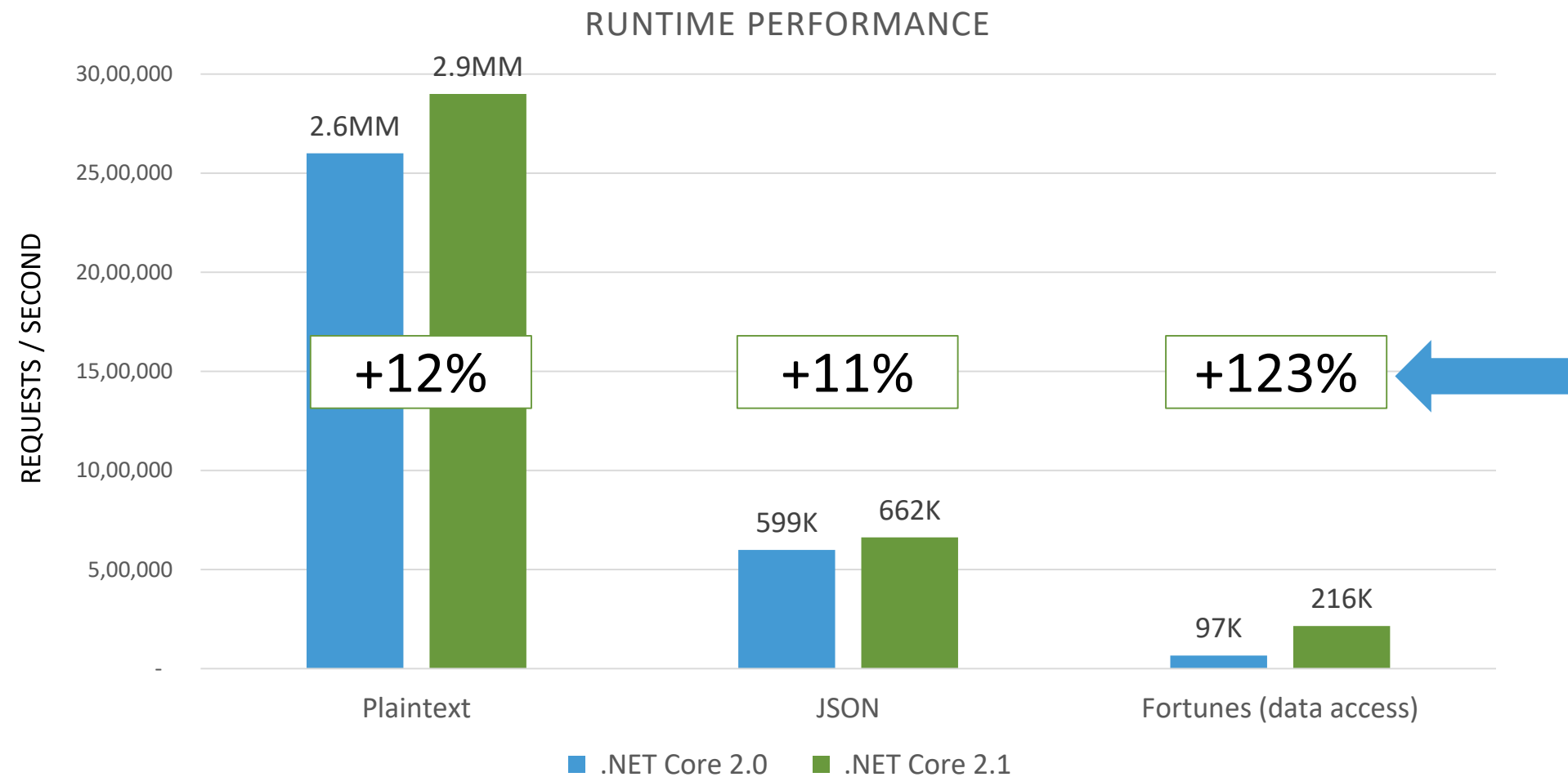
- .NET Standard 2.0
    - Brought back ~20,000 APIs
    - Supported on all the major .NET Runtimes
- . Windows Compatibility Pack
    - Brings back additional ~20,000 APIs
    - Windows only and cross platform
- NET Framework Compatibility mode
    - Target .NET Framework libraries on .NET Core

.NET Core 3.X

- Desktop workloads
  - WPF
  - WinForms

- Runtime Innovation
  - Performance
  - Evolve language + runtime together

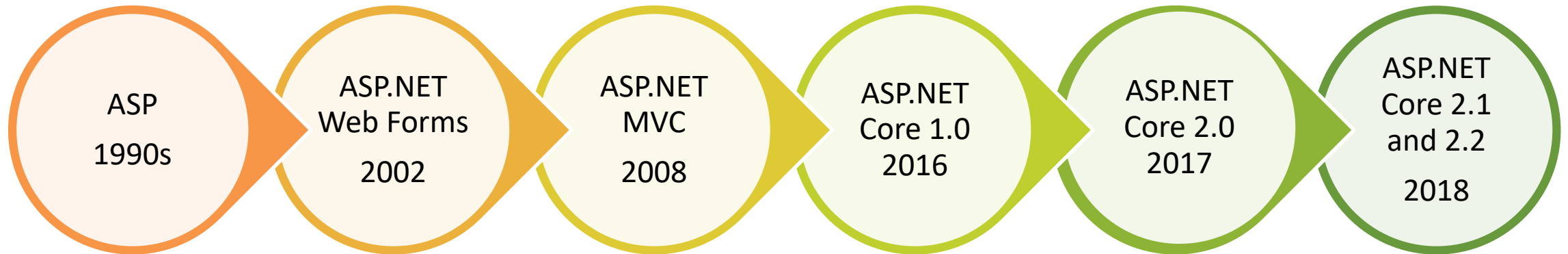Confidential For Internal Discussion Purposes Only

Mature runtime and framework

Cross platform

Open source

Customer choices

- AppLocal vs side-by-side
- Configurable
- Customizable

Fast moving

It is the future .NET

The Road to ASP.NET Core 2.2

ASP
1990s

ASP.NET
Web Forms
2002

ASP.NET
MVC
2008

ASP.NET
Core 1.0
2016

ASP.NET
Core 2.0
2017

ASP.NET
Core 2.1
and 2.2
2018

Confidential For Internal Discussion Purposes Only

infogain

Application Starting

Program class

Startup class

ConfigureServices method

Registering services

Configure method

Pipeline is created

Ready for requests

Confidential For Internal Discussion Purposes Only

Migration?

CHALLENGES

- MVC only
- C# only
- VB support started from 2.x
- New API
- New hosting model
- No IIS-centrism
- New .NET FX

OPPORTUNITIES

- X-platform
- Improved API
- Same MVC paradigm

Changes

- Tight coupling between ASP.NET and IIS
  - Presented as a big win only a few years ago

- Platform dependence
  - Windows only

- System.Web is legacy code
  - Devised back in 1997

Confidential For Internal Discussion Purposes Only

What's different?

- Configuration, startup, pipeline
  - Free IoC though ☺
  - Full rewrite of HTTP modules (including tools)
- New ways of doing old things
  - Core authentication (except Identity)
  - Embedded resources
  - Project management
- Development experience
  - Test, build, deploy, manage dependencies

Confidential For Internal Discussion Purposes Only

ASP.NET MVC

- No support for Web Forms

- Migrating code is overall doable

- Switching to ASP.NET Core mindset is easy
  - Aside a few new APIs to learn, it's all the same
  - Plus minor features in Razor

Confidential For Internal Discussion Purposes Only   27

CORE VALUES

CLIENT VALUE CREATION

Understanding and delivering what drives the value for our client partners and enable them to become high performance businesses

GROW TOGETHER

Shared commitment to openness, collaboration and empathy provides foundation for personal and professional growth.

DELIVER EXCELLENCE

Holding Accountability to work as a team to deliver outstanding quality of work to our clients, shareholders and colleagues.

INNOVATE

Challenge conventional wisdom, take risks and speak out for continuous innovation

Thank You

Infogain Corporation, HQ
485 Alberto Way Los Gatos,
CA 95032 USA
Phone: 408-355-6000
Fax: 408-355-7000

Confidential For Internal Discussion Purposes Only