

WHAT TO EXPECT FROM THE MICROSOFT .NET FRAMEWORK

VERSION 4.6



DREAM



BUILD



LAUNCH

The latest Microsoft.NET framework is not a routine upgrade. This complete re-write of .NET from the ground-up promises additional flexibility to its developers, and a lot more.

1

INTRODUCTION

2

OPEN SOURCE MICROSOFT .NET FRAMEWORK PACKAGES

- a. Availability on GitHub*
- b. Visual Studio Community*

3

THE STAND-OUT FEATURES

- a. Modularity*
- b. Improved compiler*
- c. HTTP2 support*
- d. Native integration of Roslyn code DOM compilers*
- e. Added support for asynchronous model binding*
- f. Support for code page encodings*
- g. Changes in Base Class Library (BCL)*
- h. Improvements to event tracing*

4

TECHNICAL REQUIREMENTS

5

ABOUT SUYATI

INTRODUCTION

The popular Microsoft.NET Framework, the preferred runtime environment for Windows based apps, has evolved considerably over time. Microsoft .NET 2015 RC introduces the .NET Framework 4.6 RC and .NET Core 5, with significant new features.

The Microsoft .NET runtime environment currently runs version 4.5.2. In November 2014, Microsoft launched a preview of the upcoming version 4.6, which gave developers, business managers and other enthusiasts a very good idea of what's in store. Microsoft .NET Framework 4.6 ships with Visual Studio 2015 preview and includes some exciting new features and enhancements.

What makes Microsoft.NET version 4.6 more significant than any routine upgrade is the complete re-write of .NET from the ground-up. Microsoft embarked on such a comprehensive exercise to accomplish several long-pending architectural goals, such as making it compatible with cross platform applications, increasing modularity, offering additional flexibility to its developers, and more.

OPEN SOURCE MICROSOFT .NET FRAMEWORK PACKAGES

Microsoft .NET version 4.6 underscores Microsoft's new found commitment to open source technology. That Microsoft is moving towards open source in a big way is old news; but the .NET version 4.6 confirms this trend and is an example of Microsoft actually walking the talk.

Microsoft.NET Core 5 is a completely open source stack and is capable of running on multiple operating systems. Microsoft has contributed the .NET Core 5 to the .NET foundation and placed it on GitHub. In fact, Microsoft is planning all future development only on GitHub rather than internal Microsoft infrastructure, while the .NET engineers will use the same tools that everyone else uses.

Microsoft has also launched a Visual Studio Community, a new free SKU that supports extensions for Visual Studio, for open source developers. This contains some new collection classes as well.

Developers get the best of both worlds in such a set-up. They gain access to a robust cross platform framework in open source, even as Microsoft continues to support it. Furthermore, Microsoft plans even more collaboration with the Mono Project, the free open source project aimed at establishing an Ecma standard-compliant, .NET Framework-compatible set of tools. .NET teams may now collaborate with other developers around the world even more effectively to create cloud and server based applications.

The foundation libraries that make up the Microsoft.NET core development stack are now available in GitHub, including a few .NET packages like Immutable Collections and SIMD APIs under the MIT license. The MIT license allows the licensed software to be used "without restriction". The "Immutable Collection" uses an immutable state capable of being passed freely among multiple threads. Unlike the case with read-only collections, the provider or consumer of the collection cannot change immutable collection.

THE STAND-OUT FEATURES



MODULARITY

A key strength of Microsoft.NET Core 5 and the accompanying .NET framework version 4.6 is its modular structure, which makes it possible for developers to include only those features necessary to run the app in question. The modular feature delivers the twin advantage of making the app lean and making it possible to package a private version of the .NET core framework with the app. Apps may now run on different versions of the framework on the same server, with each of that app remaining immune to changes in behavior owing to the presence of a different version of the framework for other apps.

The modular framework ships via NuGet and may be hosted on IIS or self-hosted in a custom process. Microsoft.NET Native converts source code to native code at compile time.



IMPROVED COMPILER

Just in Time compiler (JIT) compiles byte code to machine readable code at run time. The Microsoft.NET framework 4.6 introduces a new and improved 64-bit RyuJIT Compiler that delivers significant performance gains over the legacy 64 bit JIT Compiler. RyuJIT becomes open source and cross platform as well.

An area where this improvement is especially felt is startup time. The Bing team that now uses RyuJIT on some of their search-related workloads records a 25% improvement in startup time. RyuJIT co-opts SIMD improvements as well. Microsoft has created an SIMD .NET library with RyuJIT, allowing RyuJIT to optimize the SIMD types.

The new release of JIT also fixed bugs reported in the earlier versions.



HTTP2 SUPPORT

The latest version of Microsoft Visual Studio 2015 RC, of which Microsoft.NET 4.6 is a part, supports HTTP2, improving performance considerably. HTTP2 improves load time of websites by reducing latency by header compression, leveraging server-based push technology, and offering parallel loading support for page elements over a single connection. The ensuing performance boost of both existing and future Web Forms applications comes with little effort from the developer.



NATIVE INTEGRATION OF ROSLYN CODE DOM COMPILERS

Developers using Microsoft.NET version 4.6 may leverage new language features in Web Forms using the Roslyn Code DOM Compiler. Roslyn makes it explicit to developers the churning that takes place behind the scenes during the compilation process, and offers developers APIs to access such churn data. Such APIs offer valuable feedback and allow developers to optimize the code further. In .NET version 4.6, Web Forms applications use the Roslyn Code DOM compilers natively, allowing for faster compilations and reducing chances of error messages that invariably creep in when the hitherto used Code DOM compilers do not understand how to use the newer language features available, like Roslyn does.



ADDED SUPPORT FOR ASYNCHRONOUS MODEL BINDING

One major addition in Microsoft.NET 4.5 was task-based async / await methods. This feature allows developers to create asynchronous functionality within their applications easily, and populate data-specific controls such as GridViews, Repeaters and ListViews by actual methods instead of data sources. Microsoft.NET 4.6 introduces asynchronous model binding support that expands upon both the above concepts, allowing developers to decorate the various methods that govern data sources with async and await calls. The result is a more asynchronous and efficient application, and unmatched flexibility.



SUPPORT FOR CODE PAGE ENCODINGS

By default, the Microsoft.NET Core primarily supports Unicode encodings, and support for code page encodings is limited. This creates certain issues. Encoding operations face problems frequently when a Unicode character cannot be mapped to a particular code page encoding. Similarly, decoding operations go awry when invalid byte sequences cannot be translated into valid Unicode characters.

Microsoft.NET framework version 4.6 offers new support for code page encodings, sparing developers of the worry about the fallback strategy for an object encoding to be used. The `CodePagesEncodingProvider` class enables access to an encoding provider for code pages.

The Microsoft.NET Framework for the Windows Desktop supports a large set of Unicode and code page encodings. The new version allows developers to register code page encodings available in the .NET framework, but not supported in the .NET core, using the `Encoding.RegisterProvidermethod`.



CHANGES IN BASE CLASS LIBRARY (BCL)

The Base Class Library (BCL) is a collection of types in .NET Framework. Microsoft.NET 4.6 adds some new APIs that enable key scenarios in the Cross-Platform environment. For instance, the “CultureInfo.CurrentCulture” and “CultureInfo.CurrentUICulture” properties have been made read-write instead of the previous read-only, meaning that when a “CultureInfo” instance is assigned to either of these properties, both “CurrentCulture” and “CurrentUICulture” properties would update to reflect the change.



IMPROVEMENTS TO EVENT TRACING

Developers use Event Tracing for Windows (ETW), a kernel-level tracing facility that enables logging kernel or application-defined events to a log file, to debug applications or gain visibility on performance issues. The .NET Framework 4.6 enables out-of-process event ETW-based activity tracing for a larger surface area. Advanced Power Management (APM) vendors can now leverage this facility to offer lightweight tools that track the costs of individual requests and activities that cross threads, in an accurate way. This increases the performance of an application effectively. Microsoft.NET 4.6 relaxes some EventSource validation rules to allow EventSource types to implement interfaces. As such, EventSource may now be used in advanced logging systems that use interfaces to define a common logging target. It becomes possible to construct the EventSource object that logs the event directly, with the option to call one of the Write () methods to emit a self-describing event.

TECHNICAL REQUIREMENTS

The Microsoft .NET Framework 4.6 Preview is highly compatible with the earlier .NET Framework versions, such as .Net version 4, .NET version 4.5, .NET version 4.5.1 and .NET version 4.5.2, facilitating easy and seamless transition. Since the new version is modular and installs only the components applicable to a particular platform, the web installer is a much small package compared to previous packages. The web installer determines the components applicable for a particular platform automatically and downloads only those components. The minimum resources required for .NET framework 4.6 to run smoothly are 1GHz processor, 512 MB RAM, 2 and .2 B Hard Disk space (x86).

Download the Microsoft .NET Framework version 4.6 from [here](#). As of now, this is only a preview version intended for testing and feedback purposes and not for a production environment. There could be some fine tuning when the full blown production environment releases. Nonetheless, the preview gives a good idea of what is in store.

ABOUT SUYATI

Suyati provides marketing technology and integration services for companies that wish to combine the best of breed solutions and create a unified approach to customer acquisition. This unified digital marketing approach requires system integration between various CMS and CRM platforms, and a slew of ecommerce, Marketing Automation, Social Media Listening, email and social marketing, and customer service systems. Our specialized knowledge in Salesforce, open source and .Net based systems enables us to build effective custom integrated solutions for our clients.

Suyati's custom technology solutions have been deployed in companies in the US, Western Europe and Australia, and have helped many enterprises leverage the web/cloud/mobile technologies to acquire customers through integrated digital marketing. Suyati is based in Chicago with product engineering capability out of the US and India.



WWW.SUYATI.COM | SERVICES@SUYATI.COM

REFERENCE:

- <http://www.infragistics.com/community/blogs/devtoolsguy/archive/2015/04/03/what-s-new-in-microsoft-net-framework-4-6.aspx>
- <https://msdn.microsoft.com/en-us/library/ms171868%28v=vs.110%29.aspx#v46>
- <https://www.microsoft.com/en-us/download/details.aspx?id=44928>
- <http://www.c-sharpcorner.com/UploadFile/7ca517/new-features-in-net-framework-4-6-in-visual-studio-2015/>
- <http://www.devx.com/dotnet/overview-of-the-new-features-in-.net-framework-4.6.html>
- <http://blogs.msdn.com/b/dotnet/archive/2014/11/12/announcing-net-2015-preview-a-new-era-for-net.aspx> 7.
- <http://www.dotnetcurry.com/aspnet/1127/aspnet-webforms-new-features>