

What is .NET

outline

- .NET framework
- CLR
- CLI, CIL, MSIL
- ADO.NET, ASP.NET, MVC, webforms, razor
- Mono
- .NET Core
- .NET Standard

.NET framework

- Software framework
- First released early 2000's
- “consists of the common language runtime (CLR) and the .NET Framework class library (FCL)” [1]
- Provides a “code execution environment”
- Various runtime hosts for the CLR [2]
 - ASP.NET (via ISAPI filter)
 - Executable
 - Can write your own

CLR

- The virtual machine that manages execution of .NET programs [3]
- Memory
- Threads
- Code trust
- Type checking
- pinvoke
- JIT compiling to execute on current hardware

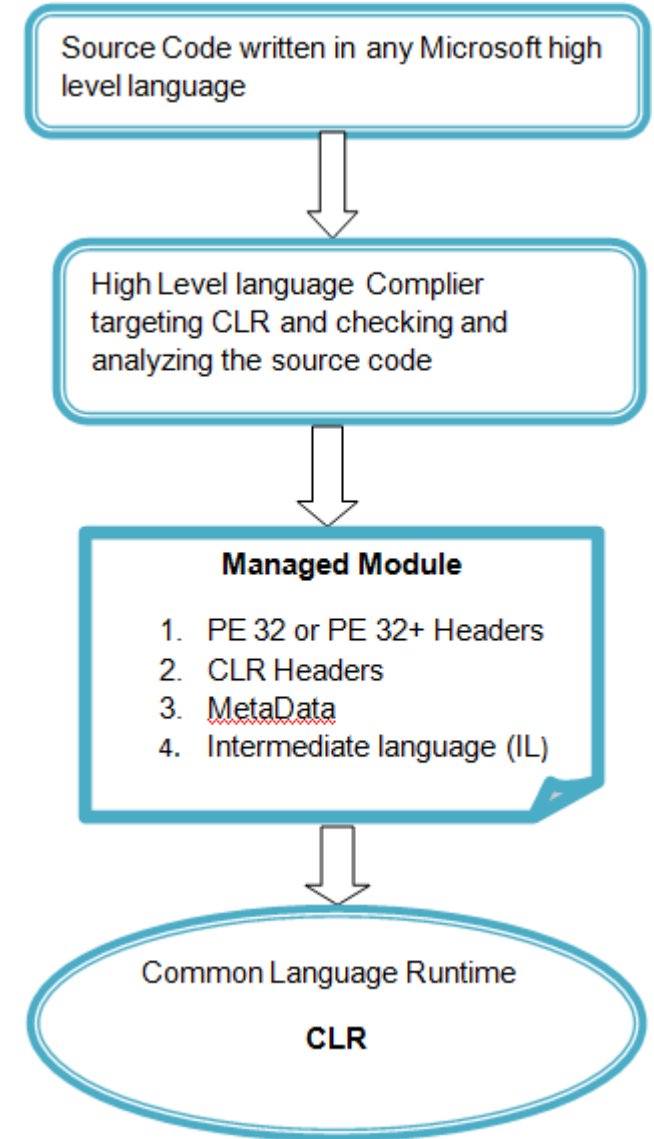
.NET Framework and CLR

- Framework and CLR are different [4]
- Can technically run different framework versions within the same CLR version
 - E.g. Windows XP only supports framework up to 4.0, running 4.5 apps is theoretically possible (with some hex editing of the binary) as long as calls are restricted to methods that were the same in 4.0 (e.g., no extensions) [5]
- Recent framework updates have been focused on
 - Optimization (including SIMD)
 - Accessories (stylus, touch screen, etc)
 - Security

Version number	CLR version	Release date	
1.0	1.0	2002-02-13	2
1.1	1.1	2003-04-24	
2.0	2.0	2005-11-07	2
3.0	2.0	2006-11-06	2
3.5	2.0	2007-11-19	2 (e
4.0	4	2010-04-12	2
4.5	4	2012-08-15	2
4.5.1	4	2013-10-17	2
4.5.2	4	2014-05-05	
4.6	4	2015-07-20	
4.6.1	4	2015-11-30 ^[34]	
4.6.2	4	2016-08-02 ^[35]	
4.7	4	2017-04-05 ^[36]	

CLR, CLI, CIL

- Common Language Infrastructure (CLI)
 - Open standard that describes executable code [6]
- Common Intermediate Language (CIL) [7]
 - Aka “IL”
 - Formerly MSIL
 - Object oriented assembly language
 - Platform independent
- A compiler takes source code and
 - Converts it to CIL using the CLI specification
 - Adds any required metadata (type information, etc)
 - This is an “assembly”



CLR, CLI, CIL

- Visual Studio tooling supports compiling C#, F#, VB.NET to CIL
- Semi C++ support [8]
- Language support include IronPython, ClojureCLR, IronScheme, others [10]
- Can write your own compiler
 - C to CIL example in F# [9]
- Ahead of time (AOT) compilation to avoid JIT
 - Has some performance benefits
 - No longer portable across hardware
- CIL makes calling a method in another CIL library convenient
 - Managed by CLR, vs unsafe dll call through pinvoke

CLR, CLI, CIL

- The CIL is available at runtime
- Metadata for the assembly needs to be available to the CLR
 - This is generally available at runtime as well
- Can use this information to modify code at runtime
 - See [21] which references [22]
- Useful for performance profiling, anti-debug avoidance
 - Other more suitable tools exist for aspect oriented programming (AOP) and mocking frameworks

.NET framework

- Includes a set of standard class libraries [11]
 - Base Class Library (BCL) [12]
 - Intrinsic types (e.g. string, datetime) and other base classes (e.g. FileStream)
 - Follows ECMA specification
 - Framework Class Library (FCL)
 - Broader library base
 - Depends on BCL
 - Provides extensions, like LINQ methods
 - System.Web, System.Data, System.Collections, System.Xml etc [13]
 - Source available at [16]
- Also includes libraries for winforms, WPF, WCF, ASP.NET, etc
 - These other libraries are mostly still proprietary (and will probably remain so)

ADO.NET, ASP.NET, MVC, webforms, razor

- ADO.NET is a .NET Framework library to interact with (MSSQL) database
 - Entity Framework is ORM for ADO.NET
- ASP.NET is a .NET framework library to serve dynamic web pages [14]
 - Typically hosted in IIS
 - Pages are “Web Forms”
 - Has “code behind” file and view file
 - Pages need to be compiled like source code

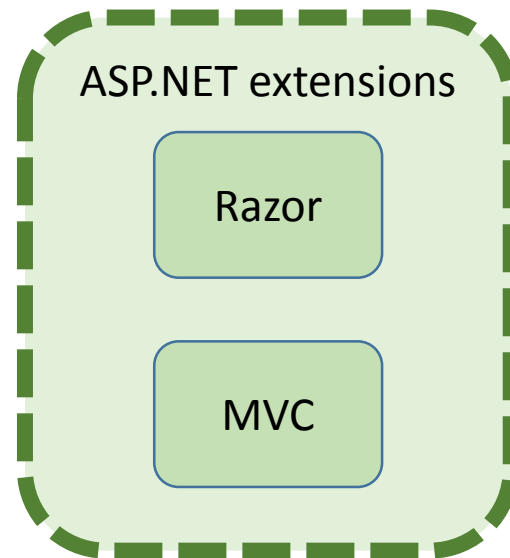
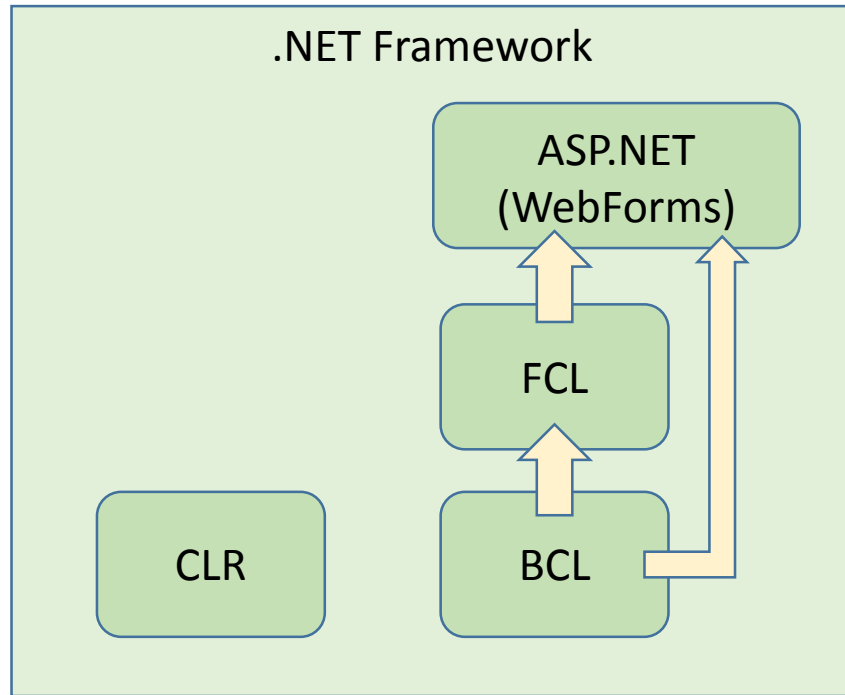
ADO.NET, ASP.NET, MVC, webforms, razor

- ASP.NET cont

- MVC is an extension to ASP.NET that comes with Visual Studio
 - Implements Model-View-Controller design pattern
 - Methods for a page are handled in the controller, or somewhere else as defined in route setup
 - Can still include code in the view itself, same as before
- Razor is typically used with MVC
 - Entire state of the webpage (“ViewState”) no longer needs to be passed around with every server call
 - Page is no longer re-rendered with every server call
 - Views are compiled on demand and can be changed at runtime
 - No more

```
if (!IsPostBack)
    // generate form
else
    // process submitted data;
```

ASP.NET visualized



- Extensions available through Visual Studio, elsewhere

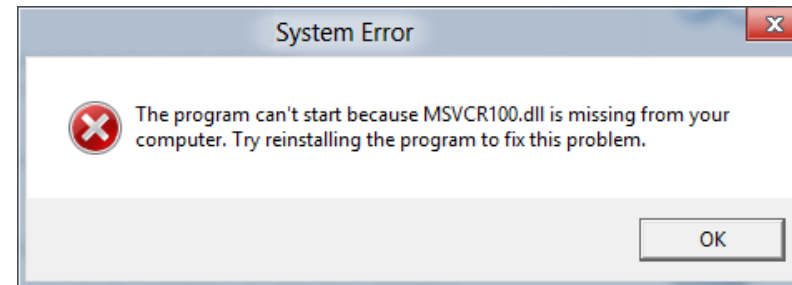
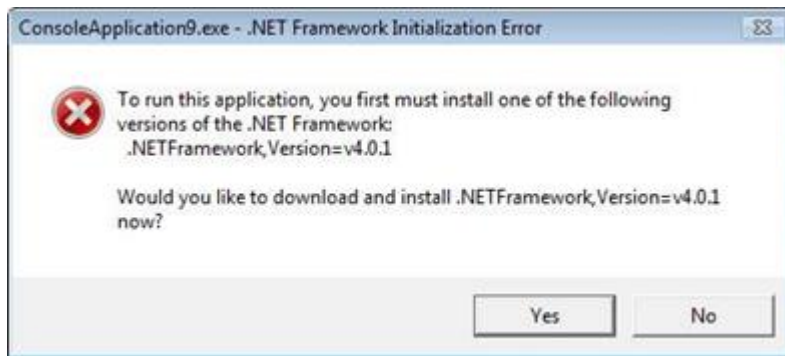
Mono

- Started as open source C# compiler and CLR [15]
- “Mono” includes the mono libraries (e.g. POSIX and OS libraries), Code Execution Engine, etc
- Alternative to .NET Framework
- MonoDevelop is the IDE

.NET core

- Cross platform, open source [17]
- Alternative to .NET framework
 - CoreCLR
 - Fork of FCL
 - ASP.NET core
 - Universal Windows Platform (UWP) libraries
- Adds additional libraries
- Missing some .NET Framework libraries
 - Windows forms, WPF
- Adds deployment to have self contained applications (via AOT compilation) [20]

- Compare to .NET Framework, or C++



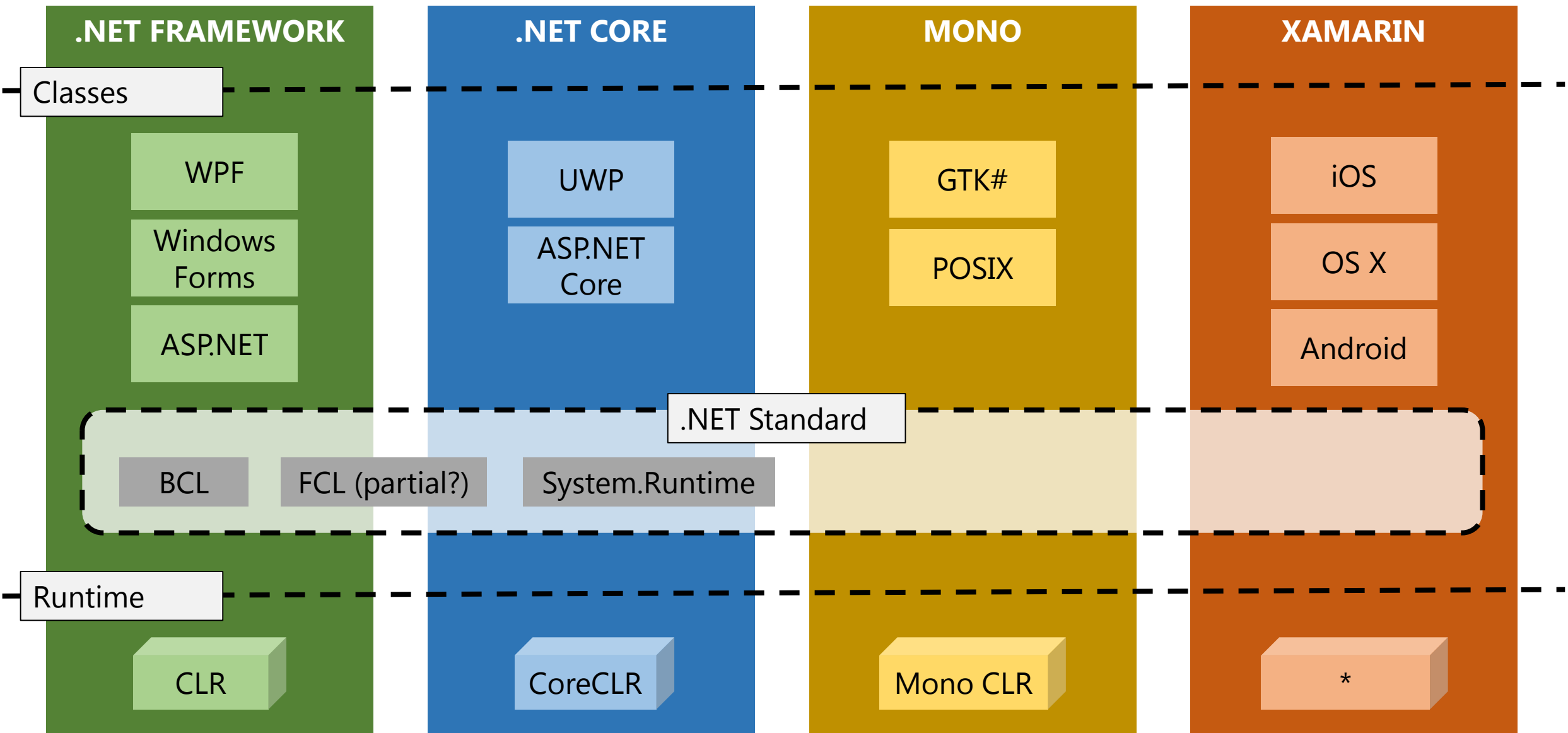
.NET Standard

- Set of API specifications (guidelines, not code)
- Supposed to be supported by multiple frameworks [18]
- Version commentary about 1.6 -> 2.0 at [19]

.NET Standard	1.0	1.1	1.2	1.3	1.4	1.5	1.6	2.0
.NET Core	1.0	1.0	1.0	1.0	1.0	1.0	1.0	2.0
.NET Framework (with tooling 1.0)	4.5	4.5	4.5.1	4.6	4.6.1	4.6.2		
.NET Framework (with tooling 2.0 preview)	4.5	4.5	4.5.1	4.6	4.6.1	4.6.1	4.6.1	4.6.1
Mono	4.6	4.6	4.6	4.6	4.6	4.6	4.6	vNext
Xamarin.iOS	10.0	10.0	10.0	10.0	10.0	10.0	10.0	vNext
Xamarin.Android	7.0	7.0	7.0	7.0	7.0	7.0	7.0	vNext
Universal Windows Platform	10.0	10.0	10.0	10.0	10.0	vNext	vNext	vNext
Windows	8.0	8.0	8.1					
Windows Phone	8.1	8.1	8.1					
Windows Phone Silverlight	8.0							

.NET Standard

- Unifying library across platforms
- Mostly based on .NET Framework mscorlib (BCL, FCL, “and friends” [19])
- Binary compatible, not necessarily behavioral compatible



- Each framework has more libraries not shown (OS specific, compatibility, etc)
- Sticking to BCL and FCL is mostly cross platform, works well for simple console apps

References

- [1] <https://docs.microsoft.com/en-us/dotnet/framework/get-started/overview>
- [2] [https://msdn.microsoft.com/en-us/library/a51xd4ze\(v=vs.71\).aspx](https://msdn.microsoft.com/en-us/library/a51xd4ze(v=vs.71).aspx)
- [3] https://en.wikipedia.org/wiki/Common_Language_Runtime
- [4] https://en.wikipedia.org/wiki/.NET_Framework
- [5] <https://stackoverflow.com/questions/17499351/is-it-possible-to-run-a-net-4-5-app-on-xp>
- [6] https://en.wikipedia.org/wiki/Common_Language_Infrastructure
- [7] https://en.wikipedia.org/wiki/Common_Intermediate_Language
- [8] <https://stackoverflow.com/questions/688386/is-c-converted-into-msil>
- [9] <http://timjones.io/blog/archive/2014/04/13/writing-a-minic-to-msil-compiler-in-fsharp-part-0-introduction>
- [10] https://en.wikipedia.org/wiki/List_of_CLI_languages
- [11] <https://stackoverflow.com/questions/807880/bcl-base-class-library-vs-fcl-framework-class-library>
- [12] https://en.wikipedia.org/wiki/List_of_data_types_of_the_Standard_Libraries#Base_Class_Library
- [13] [https://msdn.microsoft.com/en-us/library/gg145045\(v=vs.110\).aspx](https://msdn.microsoft.com/en-us/library/gg145045(v=vs.110).aspx)
- [14] <https://en.wikipedia.org/wiki/ASP.NET>
- [15] [https://en.wikipedia.org/wiki/Mono_\(software\)](https://en.wikipedia.org/wiki/Mono_(software))
- [16] <http://referencesource.microsoft.com/>
- [17] <https://docs.microsoft.com/en-us/dotnet/core/>
- [18] <https://docs.microsoft.com/en-us/dotnet/standard/library>
- [19] <https://github.com/dotnet/standard/tree/master/docs/netstandard-20>
- [20] <https://github.com/dotnet/coreclr/blob/master/Documentation/building/crossgen.md>
- [21] <https://stackoverflow.com/a/19403979/1462295>
- [22] <https://www.codeproject.com/Articles/463508/NET-CLR-Injection-Modify-IL-Code-during-Run-time>