

Laboratórios de Desenvolvimento de Software - LEI

.NET Framework

Ricardo Santos **Nuno Macedo** Ângelo Costa
`nfm@estgf.ipp.pt`

September 25, 2013



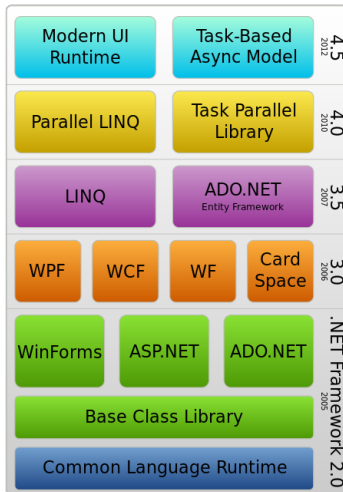
.NET Framework

- The .NET Framework is a programming framework for the Microsoft Windows platform;
- Large library of classes that provide vast array of functionalities;
- Provides language interoperability;
- Applications run over a virtual machine.

.NET History

- Development started in the late 90's under the name "*Next Generation Windows Services*"
- First releases introduced ADO.NET, ASP.NET and Windows Forms;
- Currently on version 4.5 (support for "metro-style" interfaces).

.NET Stack



The .NET Framework Stack

Wikimedia Foundation

.NET Framework Features

- Object-oriented programming:
 - Both .NET and C# are entirely based on OO;
- Good design:
 - Provides a large base class library;
- Language independence:
 - VB, C# or C++ are compiled to an intermediate language;
- Better support for dynamic web pages:
 - Introduction of ASP.NET;

.NET Framework Features

- Efficient data access:
 - ADO.NET components;
 - Built-in support for XML;
- Code sharing:
 - Code is shared through *assemblies* (rather than classic DLLs);
 - Improved security;
 - Zero-impact installation;
- Support for Web services;
- Visual Studio:
 - Fully integrated with .NET;
- C#:
 - Simple, modern and general-purpose language.

Connection with C#

- C# was designed specifically for .NET;
- The architecture of C# reflects that of the underlying .NET;
- It is an independent language, not part of .NET.

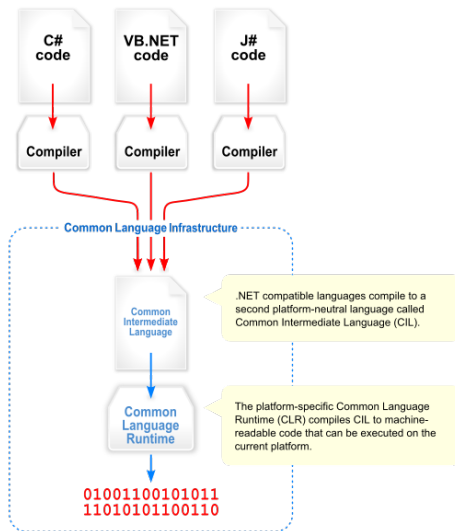
Common Language Infrastructure

- *CLI* is an open specification by Microsoft describing how to execute applications in .NET;
- Specifies exception handling, garbage collection, security and interoperability;
- Microsoft's own implementation for Windows is the *Common Language Runtime*;
- There are open source implementations for other OS.

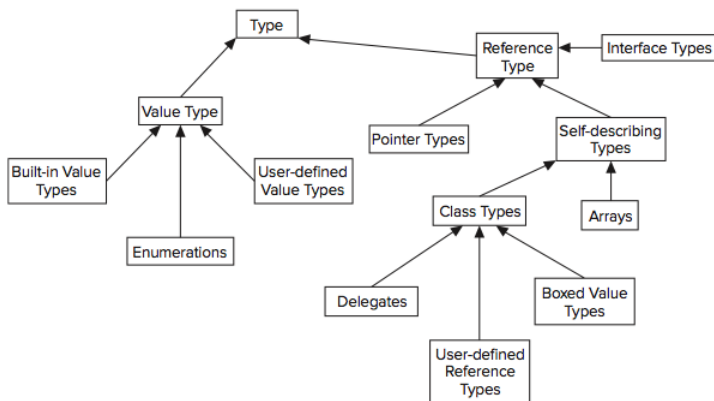
Common Language Infrastructure

- *Common Type System* (CTS):
 - Data types that must be shared by all CLI-compliant languages;
- *Common Language Specification* (CLS):
 - A set of rules that all CLI-compliant languages must follow;
- *Common Intermediate Language* (CIL):
 - CLI-compliant languages compile to CIL;
 - CIL is then compiled to platform-specific code.

Common Language Infrastructure



Common Type System



Common Language Infrastructure

- *Common Language Runtime* (CLR) is the .NET runtime execution environment developed by Microsoft;
- Applications are run over CLR rather than directly over the hardware;
 - Code is first compile to an CIL;
 - CLR compiles CIL to platform specific code.

Common Language Infrastructure

- Platform independence:
 - Code compiled to CIL can be run in any platform with the .NET framework;
 - Full implementation only available for Windows (although there are open-source implementations for other platforms);
- Performance:
 - CIL is compiled *just-in-time*, instead of compiling the whole application;
 - Could theoretically be more efficient, since it can take advantage of the features of the particular processor;
- Language interoperability:
 - Code compiled to CIL from different languages is interoperable.

Microsoft Intermediate Language

- Features of Microsoft CIL:
 - Object orientation and the use of interfaces;
 - Strong distinction between value and reference types;
 - Strong data typing;
 - Error handling using exceptions;
 - Use of attributes.
- CIL-compliant languages inherit these features.

Strong Data Typing

- All variables must be assigned a concrete type;
- Promotes language interoperability: CTS and CLS;
- Garbage collection;
- Security;
- Application domains:
 - Memory of a single process may be divided for different applications.

Garbage collection

- Objects are allocated in the memory heap (memory managed by CLR);
- When the heap needs cleaning, the garbage collector runs through all variables;
- If an object is not referenced by any variable, it is removed.

Assemblies

- Compiled code units of .NET;
- Can be executable (EXE) or libraries (DLL);
 - Not the same as the native EXE and DLL;
- Assemblies contain meta-data regarding versioning and security:
 - No need to access the registry.

Base Class Library

- Massive collection of libraries designed to be easy to use;
- Include:
 - Core features provided by CIL (primitive types from the CTS);
 - User interface (Windows Forms and WPF);
 - Web functionalities (ASP.NET);
 - Data access (ADO.NET, SQL Server);
 - File system and registry access;
 - Networking and web browsing;
 - Access to aspects of the Windows OS;
 - ...

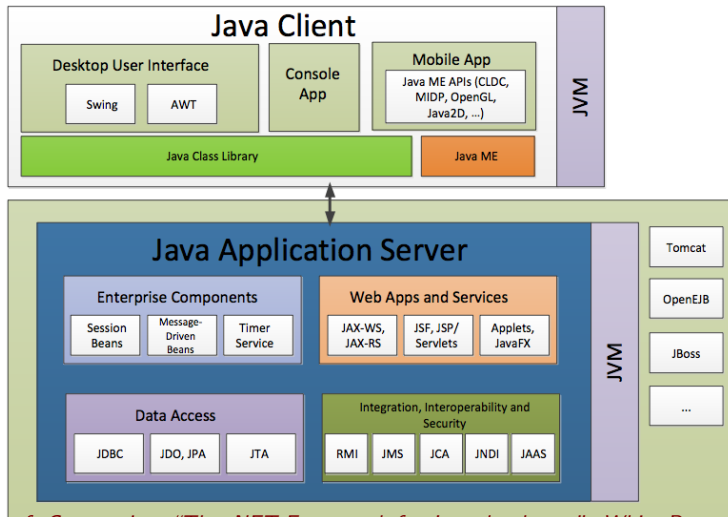
.NET vs. Java EE

- Java compiles to machine-independent byte code;
- Run over JVM;
- JDK is implemented in virtually every OS;
- A wide variety of tool support;

.NET vs. Java EE

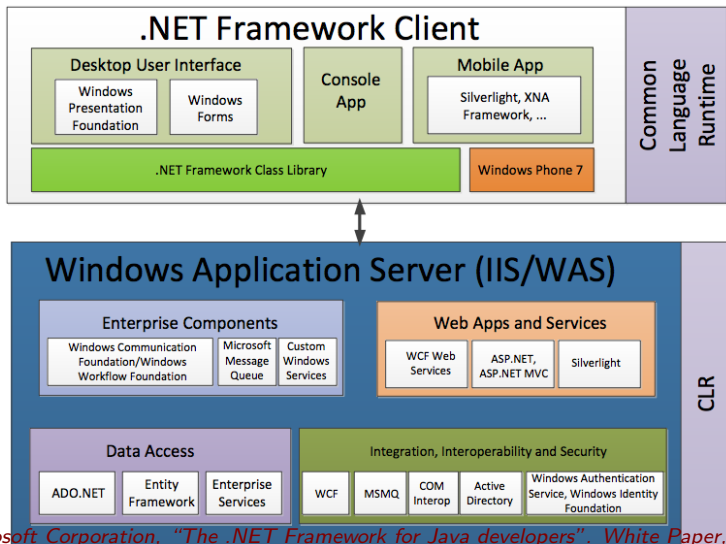
- Any compliant language compiles to CLR;
- Run over CLR;
- CLR available on Windows;
- Strongly tied to Visual Studio;

.NET vs. Java EE



Microsoft Corporation, "The .NET Framework for Java developers". White Paper, 2011.

.NET vs. Java EE



Microsoft Corporation, "The .NET Framework for Java developers". White Paper, 2011.

Bibliography



Nagel, C., Evjen, B., Glynn, J., Watson, K. and Skinner, M.
Professional C# 4 and .NET 4.
Wrox, 2010.