

Công nghệ .net



ThS. Nguyễn Thị Loan
Khoa CNTT - DQN

Chương 1. GIỚI THIỆU VỀ CÔNG NGHỆ .NET

- Giới thiệu .NET Platform
- Giới thiệu .NET Framework
- Những thuận lợi của .NET Framework
- Các thành phần của .NET Framework
- Các loại ngôn ngữ lập trình trên .NET
- Các loại ứng dụng có thể viết với .NET
- Quá trình tạo, biên dịch và chạy chương trình

2

.NET Platform

- .NET Platform
 - .NET Platform là một tập các công cụ và công nghệ giúp xây dựng dễ dàng các ứng dụng trên Internet.
- Mục đích của .NET Platform
 - Đơn giản hóa việc phát triển các ứng dụng trong môi trường phân bố của Internet

3

.NET Platform

- Các thành phần của .NET Platform
 - .NET Framework
 - .NET My Services
 - .NET Enterprise Servers
 - Visual Studio .NET

4

.NET Platform

- .NET Framework
 - Microsoft .NET Framework là môi trường chung cho việc xây dựng, triển khai và chạy ứng dụng
 - Không gắn chặt vào hệ điều hành
 - Không gắn chặt vào ngôn ngữ
 - Nhiều tiện ích khác giúp cho việc phát triển các ứng dụng nhanh, hiệu quả hơn.

5

.NET Platform

- .NET My Services
 - .NET My Services là một tập XML Web Service cho phép người dùng truy cập thông tin qua Internet. Dùng các .NET My Service các ứng dụng có thể truyền thông trực tiếp bằng giao thức SOAP và XML
 - Một số services
 - .NET Calendar,
 - .NET Contacts,
 - .NET Inbox,
 - .NET Documents
 - ...

6

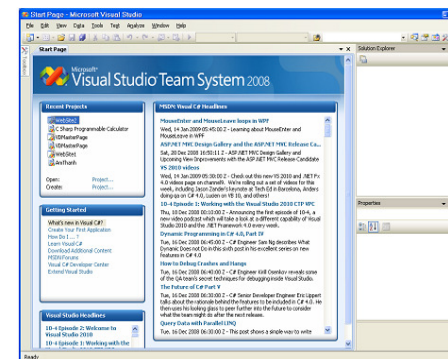
.NET Platform

- .NET Enterprise Servers
 - Một nhóm các sản phẩm của Microsoft. Mỗi sản phẩm trong họ này phụ trách một thành phần then chốt cho kiến trúc ứng dụng đang phát triển
 - Một số sản phẩm
 - Microsoft SQL Server
 - Microsoft Biztalk Server
 - Microsoft Host integration Server
 - ...

7

.NET Platform

- Visual Studio .NET
 - VS.NET là môi trường phát triển để xây dựng các ứng dụng trên .NET Framework.



8

.NET Framework

- Các mục đích thiết kế .NET Framework
 - Mô hình lập trình thống nhất
 - Đơn giản việc phát triển ứng dụng
 - Môi trường thực thi code an toàn
 - Độc lập ngôn ngữ (hỗ trợ nhiều ngôn ngữ lập trình)
 - Hỗ trợ XML Web Service (Dựa trên các công nghệ chuẩn: HTTP, XML, SOAP)

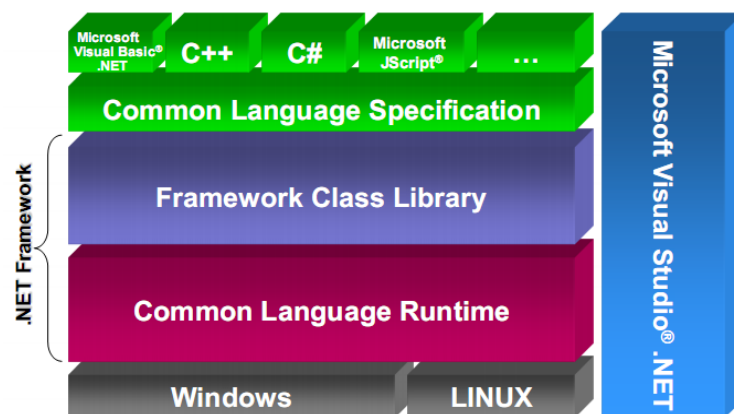
9

.NET Framework

- Các phiên bản .NET Framework
 - .NET Framework 1.0 – Visual Studio 2002
 - .NET Framework 1.1 – Visual Studio 2003
 - .NET Framework 2.0 – Visual Studio 2005
 - .NET Framework 3.0 – Visual Studio 2005
 - .NET Framework 3.5 – Visual Studio 2008
 - .NET Framework 4.0 – Visual Studio 2010
 - .NET Framework 4.5 – Visual Studio 2013
- Các thành phần trong .NET Framework: .NET Framework gồm có 2 thành phần chính:
 - Common Language Runtime (CLR)
 - Framework Class Library (FCL)

10

.NET Framework



11

.NET Framework

- CLR làm nhiệm vụ quản lý sự thực thi mã lệnh và tất cả các tác vụ liên quan đến nó: biên dịch, quản lý bộ nhớ, bảo mật, quản lý tuyến đoạn, và thực thi an toàn kiểu.
- FCL là thư viện kiểu dữ liệu có thể tái sử dụng (gồm các class, structure, ...) dành cho các ứng dụng thực thi trong .NET. Tất cả các ngôn ngữ hỗ trợ .NET Framework đều sử dụng thư viện lớp dùng chung này.

12

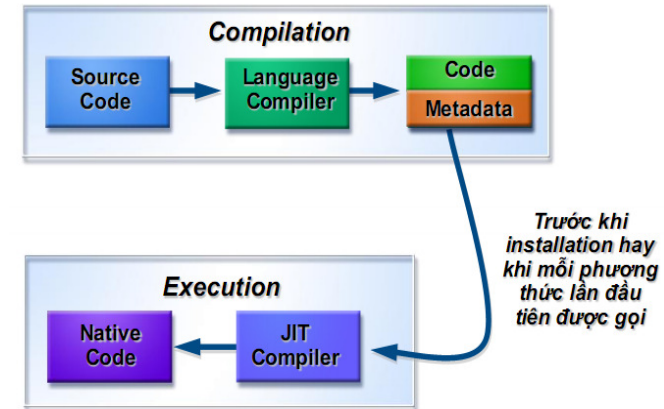
Common Language Runtime

- Common Language Runtime (CLR)
 - CLR là môi trường thực thi cho ngôn ngữ chung và các dịch vụ trong quá trình thực thi
- Một số dịch vụ CLR
 - Biên dịch
 - Quản lý bộ nhớ
 - Quản lý thread
 - Quản lý ngoại lệ
 - Bảo mật
 - Kiểm tra an toàn kiểu
 - ...

13

Common Language Runtime

- Biên dịch và thực thi



14

Common Language Runtime

- Biên dịch và thực thi:
 - Chương trình nguồn trước hết được biên dịch và đóng gói thành một khối gọi là assembly. Khối này sẽ chứa các mã lệnh ngôn ngữ trung gian và các metadata mô tả thông tin cần thiết cho sự hoạt động của khối.
 - Mỗi khi có yêu cầu thực thi assembly nói trên, CLR sẽ chuyển đổi mã lệnh ngôn ngữ trung gian trong assembly thành mã lệnh tương thích với CPU cụ thể trước khi có thể thực thi.

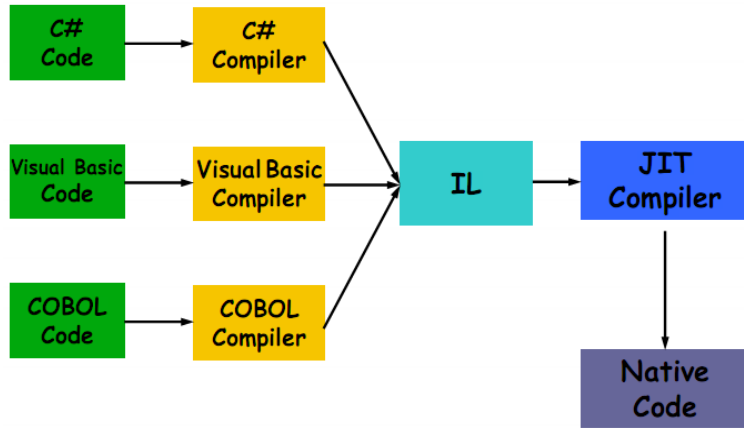
15

Common Language Runtime

- Các loại code trong .NET
 - Source code
 - MSIL (Microsoft Intermediate Language – Ngôn ngữ trung gian của Microsoft)
 - MSIL là tập các lệnh, mã lệnh cấp thấp do Microsoft sáng tạo ra.
 - Các source code được viết bằng nhiều loại ngôn ngữ khác nhau đều phải dịch sang MSIL
 - OptIL: là tập con của MSIL có thể được sinh bằng trình biên dịch tối ưu. OptIL giúp cho việc sinh nhanh native code
 - Native machine code (mã máy thuần túy)

16

Common Language Runtime



17

Common Language Runtime

■ Assembly

- Một assembly được tạo ra khi trình biên dịch tương thích với .NET chuyển một file chứa mã nguồn thành một file DLL hoặc EXE.
- Một assembly chứa một manifest, metadata, và ngôn ngữ trung gian sinh bởi trình biên dịch cụ thể.
- Một assembly có thể chứa một tập hợp gồm một hay nhiều file chứa phần mã lệnh hoặc tài nguyên (như ảnh hoặc dữ liệu XML).

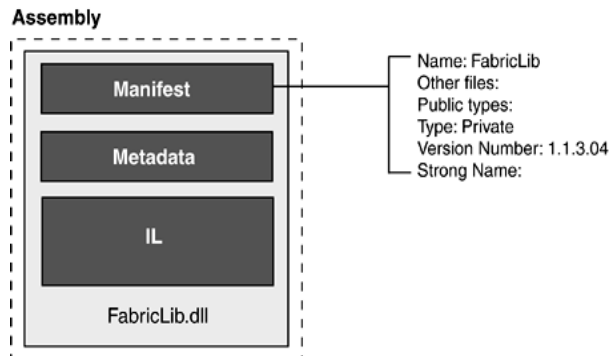
■ Phân loại Assembly

- Process assemblies (EXE)
- Library assemblies (DLL).

18

Common Language Runtime

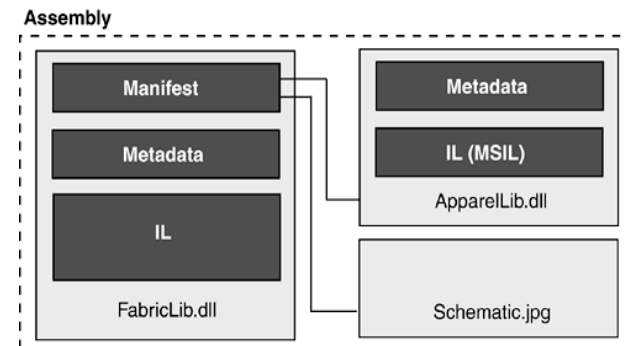
■ Các thành phần trong assembly



19

Common Language Runtime

■ Một assembly chứa nhiều file



20

Common Language Runtime

■ Manifest

- Là một tập hợp các bảng chứa các metadata trong đó liệt kê tên của tất cả các file trong assembly, tham chiếu đến các assembly bên ngoài, và các thông tin như tên, phiên bản để định danh assembly đó.
- Khi một assembly được nạp, nhiệm vụ đầu tiên của CLR là mở file chứa manifest để có thể định danh các thành viên có trong assembly.

21

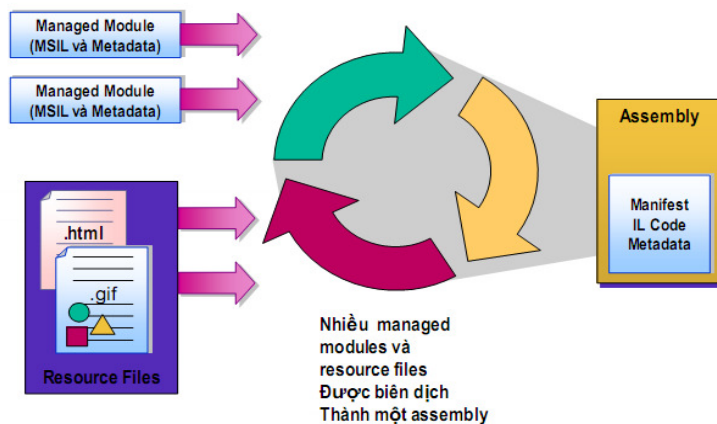
Common Language Runtime

■ Metadata

- Metadata là thông tin nhị phân dùng để mô tả IL code.
- Khi code được thực thi, runtime nạp metadata vào memory và các tham chiếu đến nó để khám phá các thông tin về code của chương trình như: các class, các thành viên, thừa kế,
- ...

22

Common Language Runtime



23

Common Language Runtime

■ JIT Compiler

- Trình biên dịch JIT là trình biên dịch lúc thực thi với nhiệm vụ chuyển đổi MSIL (assemblies) sang native code.
- Đặc điểm cơ bản của JIT
 - JIP độc lập ngôn ngữ lập trình
 - JIT phụ thuộc kiến trúc CPU
 - JIP biên dịch cho mỗi phương thức lần đầu tiên chạy
 - Native code được lưu lại để dùng lại sau này
- CLR cung cấp 3 trình biên dịch JIT để chuyển MSIL sang native code:
 - EconoJIT
 - JIT
 - optJIT

24

.NET Framework Class Library

- Framework Class Library – FCL: Một tập thư viện lớp trong .NET
- .Net Framework thay thế phần lớn (mặc dù không phải tất cả) các tập Windows API bằng thư viện lớp hướng đối tượng và được quản lý tốt

25

Framework Class Library



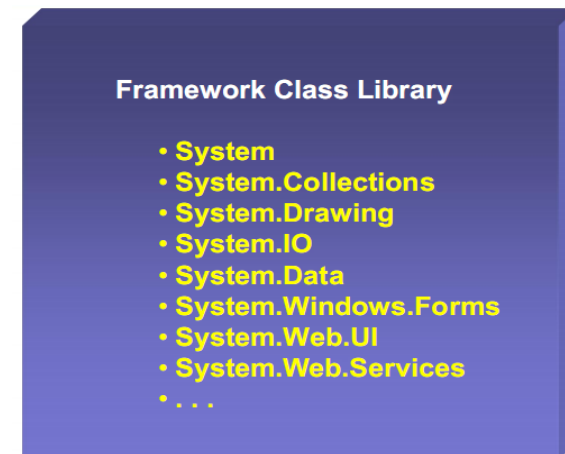
26

Framework Class Library

- Các lớp trong FCL có thể được chia thành các loại sau
 - Hỗ trợ chức năng chính: tương tác với các kiểu dữ liệu cơ bản và collection; console, network và file I/O; và tương tác dễ dàng với các runtime.
 - Hỗ trợ tương tác cơ sở dữ liệu; dùng và sinh dữ liệu XML; và cấu trúc dữ liệu.
 - Hỗ trợ xây dựng các ứng dụng web.
 - Hỗ trợ xây dựng các ứng dụng desktop
 - Hỗ trợ xây dựng các ứng dụng XML web service

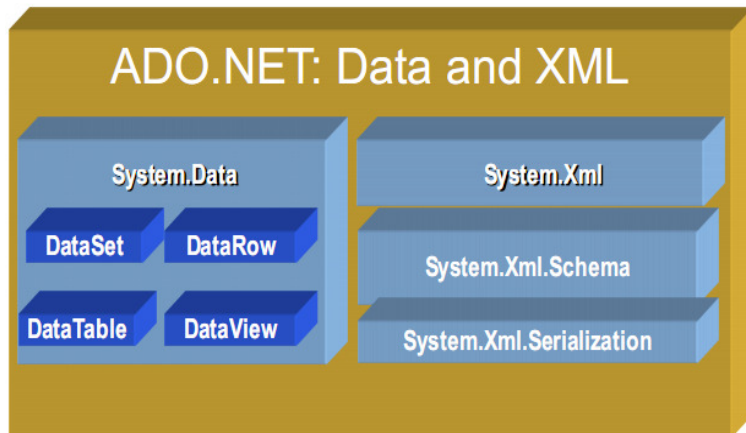
27

Framework Class Library



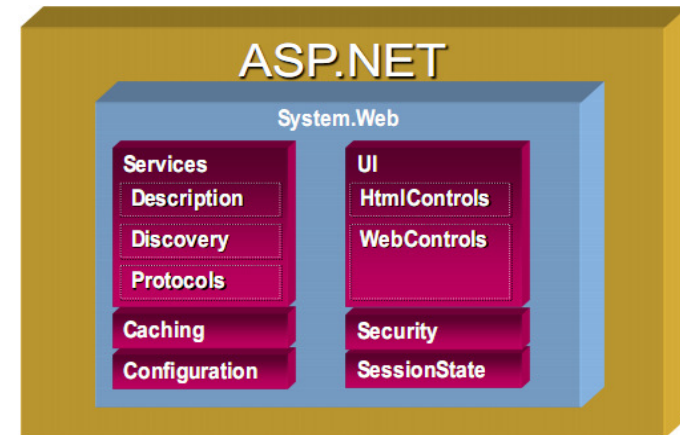
28

ADO.NET: Data và XML



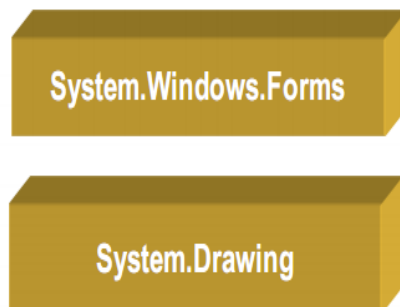
29

Web Forms và XML Web Services



30

User Interface cho Windows



31

Các ngôn ngữ trong .NET Framework

- **C# (C-Sharp)** – được thiết kế cho .NET
Ngôn ngữ hướng thành phần mới (component-oriented)
- **C++ .NET**
Các mở rộng code được quản lý cho C++
- **Visual Basic .NET**
Version mới của Visual Basic, có nhiều đổi mới
- **JScript .NET**
Version mới của Jscript, tăng cường sự thực thi
- **J# .NET**
Ngôn ngữ Java cho .NET
- **Ngôn ngữ hãng thứ ba (Third-party)**
APL, COBOL, Pascal, Eiffel, Haskell, ML, Oberon, Perl, Python, Scheme, Smalltalk...

32

Các loại ứng dụng

- Desktop application
 - Windows Form
 - Console Application
- Web application (ASP.NET)
- Services
 - ASP.NET Web service
 - Windows Service
- Library
 - Class library
 - Windows control library
 - Web control library

33

Các bước phát triển ứng dụng

- Viết mã nguồn bằng 1 trong các ngôn ngữ được .NET hỗ trợ (C#/ VB.NET, ...)
- Dịch mã nguồn sang file *.dll hay *.exe chứa mã trung gian bằng chương trình csc.exe
- Khi chương trình được chạy trong CLR, JIT sẽ dịch các mã trung gian sang native code và thực thi
- Native code được thực thi trong ngữ cảnh của CLR

34