

# Assemblies

Phạm Thi Vương

# Nội dung chính



Tổng quan Assembly



Cấu trúc Assembly



Hỗ trợ xuyên ngôn ngữ



Shared Assembly

# Nội dung chính



**Tổng quan Assembly**



**Cấu trúc Assembly**



**Hỗ trợ xuyên ngôn ngữ**



**Shared Assembly**



# Tổng quan Assembly

- Tiền thân của Assembly là các thư viện DLL
- Tuy nhiên, khi sử dụng DLL thường xảy ra vấn đề về tương thích phiên bản

# Tổng quan Assembly

- Assembly là đơn vị cài đặt tự mô tả
- Gồm một hoặc nhiều file
- Một assembly có thể bao gồm các file tài nguyên, metadata, DLL, EXE
- Assembly có thể private hoặc shared

# Tổng quan Assembly

- Tính năng của Assembly
  - Tự mô tả
  - Version dependencies
  - Có thể được load side-by-side
  - Sử dụng Application Domain
  - No-touch deployment



# Nội dung chính



Tổng quan Assembly



**Cấu trúc Assembly**



Hỗ trợ xuyên ngôn ngữ



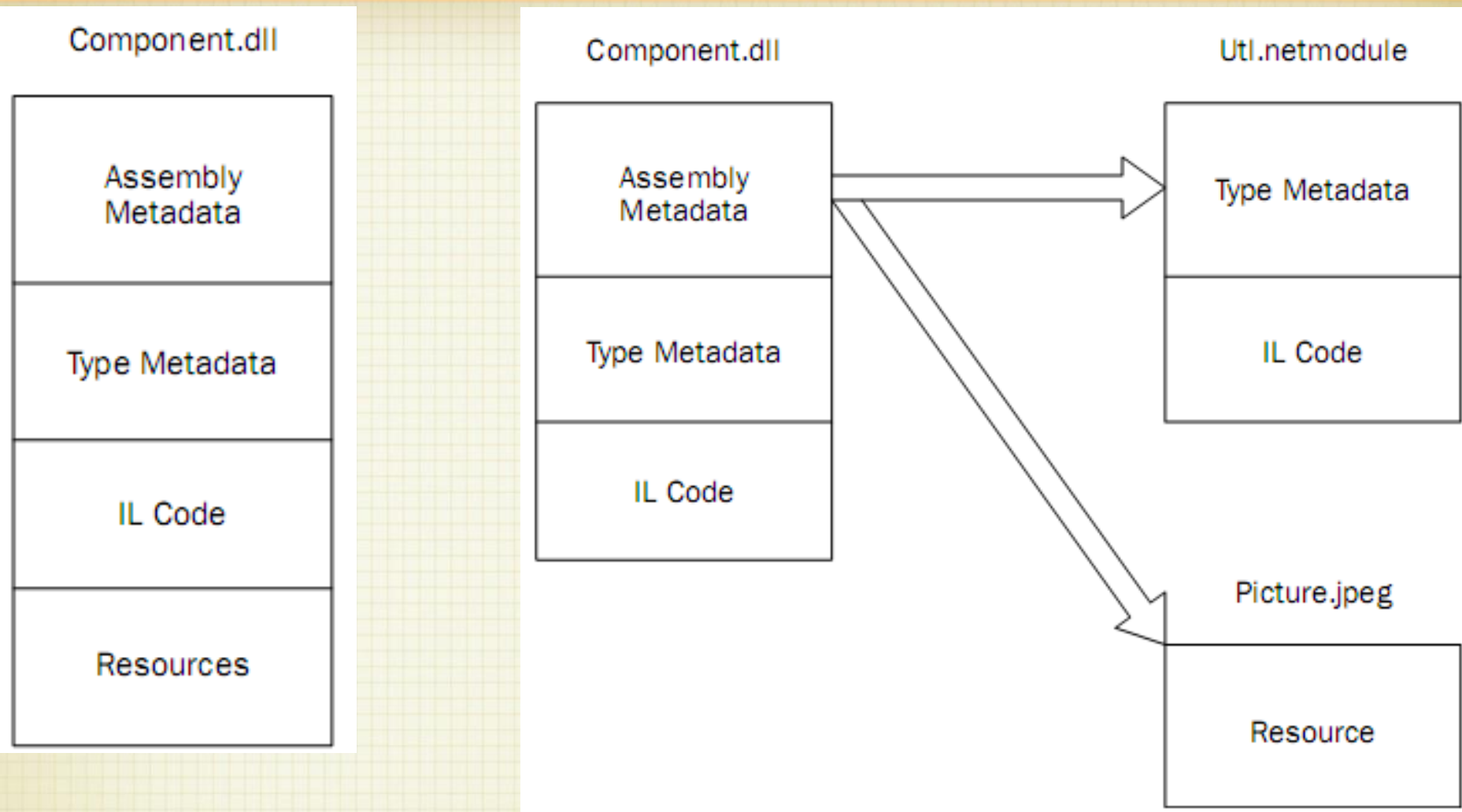
Shared Assembly

# Cấu trúc Assembly

- Một Assembly bao gồm:
  - Assembly metadata
  - Type metadata
  - Mã MSIL
  - Các tài nguyên
- Tất cả các thành phần có thể nằm trong một hoặc nhiều file
- Toàn bộ assembly là một đơn vị cài đặt

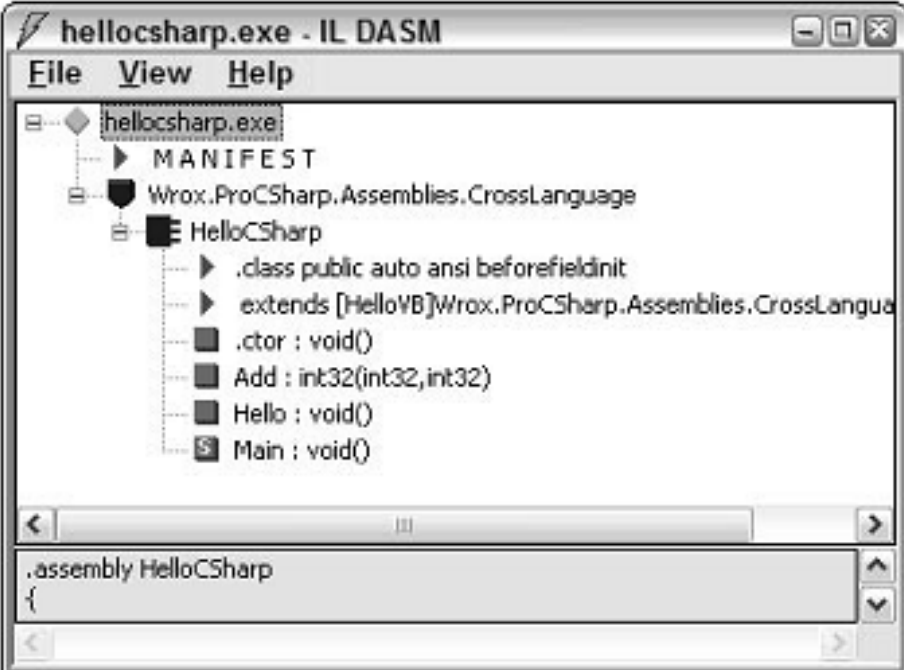


# Cấu trúc Assembly

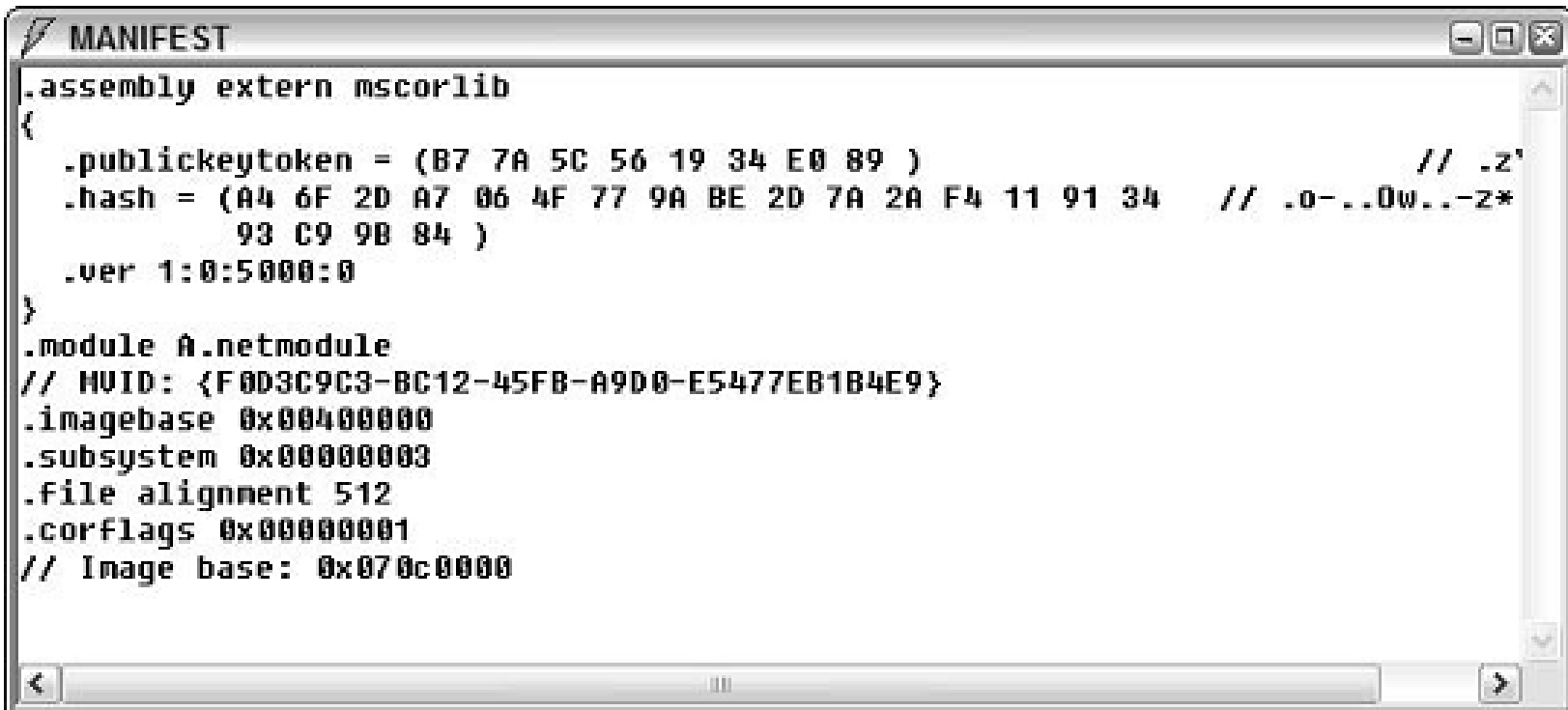


# Assembly Manifest

- Mô tả assembly với các thông tin cần tham chiếu và tất cả các dependency
- Cấu trúc Assembly Manifest bao gồm:
  - Identity (tên, phiên bản, culture, public key)
  - Danh sách các file thuộc về assembly
  - Danh sách các assembly được tham chiếu
  - Tập hợp các yêu cầu permission
  - Những kiểu được export



Sử dụng ILDASM.exe để xem thông tin assembly





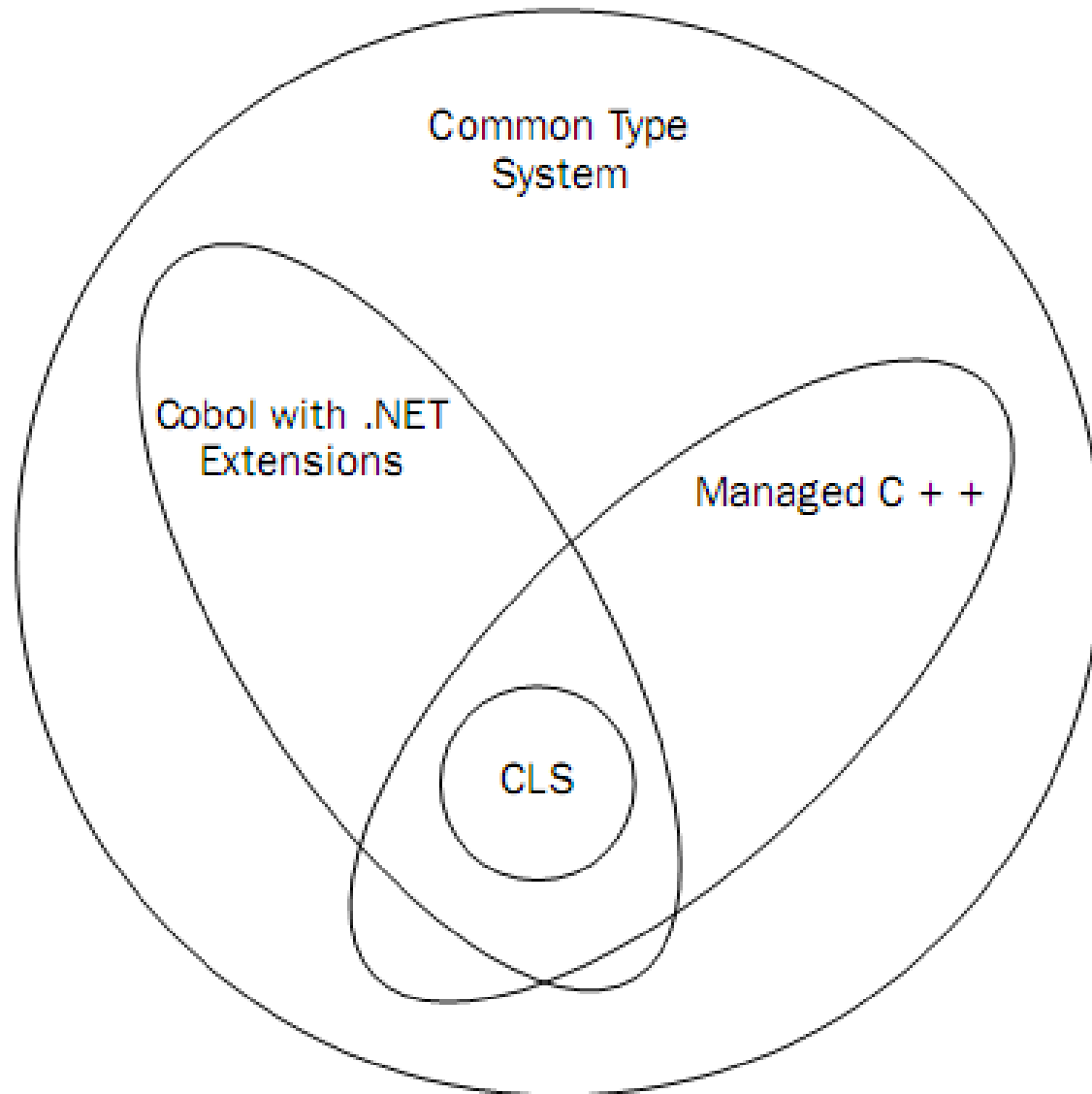
# Nội dung chính

- 1 Tổng quan Assembly
- 2 Cấu trúc Assembly
- 3 Hỗ trợ xuyên ngôn ngữ
- 4 Shared Assembly

# CTS vs CLS

- CTS định nghĩa một tập các luật mà trình biên dịch phải tuân theo để định nghĩa, tham chiếu, sử dụng và lưu trữ những kiểu trị và tham chiếu
- Để xây dựng những thành phần có thể sử dụng được trên tất cả các ngôn ngữ của .NET, ta phải sử dụng CLS

# CTS vs CLS





# Common Language Specification

- CLS định nghĩa những yêu cầu sao cho các thành phần thỏa CLS compliant: có thể được sử dụng trên những ngôn ngữ .NET khác

# Common Language Specification

- Các luật CLS (Chỉ cần áp dụng cho các thành viên public và protected):
  - Tất cả các kiểu xuất hiện trong phương thức phải thỏa CLS-Compliant
  - Những phần tử mảng phải có kiểu thỏa CLS-Compliant. Mảng được sử dụng theo 0-indexed
  - Một lớp CLS-Compliant phải kế thừa từ một lớp CLS-Compliant (System.Object là 1 lớp CLS-Compliant)

# Common Language Specification

- Các luật CLS (Chỉ cần áp dụng cho các thành viên public và protected):
  - Mặc dù tên phương thức trong các lớp CLS-Compliant không phân biệt chữ hoa/thường, các tên phương thức có thể không chỉ khác nhau ở chữ hoa/thường
  - Các Enumeration phải là kiểu Int16, Int32 hoặc Int64



# CLSCompliant Attribute

- Để compiler cảnh báo khi một phương thức public hoặc protected không thỏa Compliant, ta thiết lập attribute CLSCompliant trong file AssemblyInfo.cs

```
[assembly: System.CLSCompliant(true)]
```
- Để cài đặt một phương thức không thỏa CLSCompliant trong một assembly compliant, ta đánh dấu attribute Compliant của assembly đó là false

```
[CLSCompliant(false)]  
void Method(uint i)  
{...}
```

# Nội dung chính



Tổng quan Assembly



Cấu trúc Assembly



Hỗ trợ xuyên ngôn ngữ



**Shared Assembly**

# Shared Assembly

- Còn được gọi là Global Assembly
- Là public assembly được dùng chung bởi nhiều ứng dụng
- Không chứa trong thư mục bin của ứng dụng mà được đặt trong GAC (Global Assembly Cache)



# Global Assembly Cache (GAC)

- Là cache được dùng cho các Global Assembly
- Hầu hết global assembly và một số private assembly được cài đặt trong GAC
- Nếu một private assembly được biên dịch sang native code bằng Native Image Generator thì native code cũng nằm trong GAC

Computer > WIN 7 (C:) > Windows > assembly

Organize Open Share with Burn New folder

★ Favorites

Desktop

Downloads

Recent Places

Libraries

Documents

Music

Pictures

Videos

Homegroup

Computer

WIN 7 (C:)

DATA (D:)

SOFT (E:)

Assembly Name	Version	Cul...	Public Key Token	Proces...
Accessibility	2.0.0.0		b03f5f7f11d50a3a	MSIL
ADODB	7.0.330...		b03f5f7f11d50a3a	
AspNetMMCEExt	2.0.0.0		b03f5f7f11d50a3a	MSIL
AuditPolicyGPMana...	6.1.0.0		31bf3856ad364e35	x86
BDATunePIA	6.1.0.0		31bf3856ad364e35	x86
ComSvcConfig	3.0.0.0		b03f5f7f11d50a3a	MSIL
cscompmgd	8.0.0.0		b03f5f7f11d50a3a	MSIL
CustomMarshalers	2.0.0.0		b03f5f7f11d50a3a	x86
dao	10.0.45...		31bf3856ad364e35	
dfsvc	2.0.0.0		b03f5f7f11d50a3a	MSIL
DTEParseMgd	10.0.0.0		89845dcd8080cc91	x86
ehCIR	6.1.0.0		31bf3856ad364e35	MSIL
ehexthost	6.1.0.0		31bf3856ad364e35	MSIL
ehiActivScp	6.1.0.0		31bf3856ad364e35	MSIL
ehiBmlData				
ehiExtens				
ehiTV	6.1.0.0		31bf3856ad364e35	MSIL
ehiProxy	6.1.0.0		31bf3856ad364e35	MSIL
ehiTV	6.1.0.0		31bf3856ad364e35	MSIL

ehiActivScp, Version=6.1.0.0, Culture=neutral,  
PublicKeyToken=31bf3856ad364e35, processorArchitecture=MSIL

assembly Date modified: 12/15/2011 1:36 PM  
File folder Date created: 7/14/2009 9:37 AM

# Strong name

- Một shared assembly phải có một strong name để nhận dạng duy nhất assembly
- Strong name được cấu thành từ:
  - Tên của chính assembly
  - Một *Version number*
  - Một *Public key*
  - Một *Culture*



# Tạo Shared Assembly

1. Tạo một strong name
2. Liên kết strong name với assembly
3. Đặt assembly trong GAC

# Tạo Strong Name

- Vào command prompt bằng **Microsoft .NET Framework SDK v2.0 -> SDK Command prompt**
- Vào **c:\csharp\counterlibrary.**
- **sn -k srikanth.key** (Viết cặp public key và private key vào file srikanth.key)

# Liên kết Strong Name với Assembly

- Mở project properties
- Trong tab Signing check ô **Sign the assembly**
- Chọn file chứa key trong **Choose a strong name key file**
- Build project

Sau khi build, assembly sẽ được liên kết với public key và được đóng dấu bởi private key



# Đặt Assembly trong GAC

- Sử dụng tool GACUTIL

```
c:\csharp\counterlibrary\bin\Release>gacutil -i counterlibrary.dll
```

# Thank You !

[www.themegallery.com](http://www.themegallery.com)