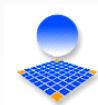


Introduction à .NET

Clémentine Nebut
LIRMM / Université de Montpellier 2



C'est quoi .NET ?

".NET is the Microsoft Web services strategy to connect information, people, systems, and devices through software."

<http://www.microsoft.com/net/basics.mspix>

"The .NET Framework is a development and execution environment that allows different programming languages & libraries to work together seamlessly to create Windows-based applications that are easier to build, manage, deploy, and integrate with other networked systems."

<http://www.msdn.microsoft.com/netframework/gettingstarted/default.aspx>

Intro .net Archi .net Langages CLR BCL Conclu

C'est quoi .NET ?

- La réponse de Microsoft à J2EE ...
- Une plateforme multi-langages
- Un ensemble de classes de bibliothèque
- Des facilités :
 - pour créer et utiliser des services web
 - et globalement pour la création d'applis n-tiers
- Un langage dédié : C#

Intro .net Archi .net Langages CLR BCL Conclu

.NET 1, 2, 3

- .net 1.0
 - avril 2003
- .net 2
 - 2.0 : janvier 2006
- .net 3.0 : WCF, WPF, WF, Cardspace
 - 3.0 : novembre 2006
 - 3.5 beta 2 : août 2007



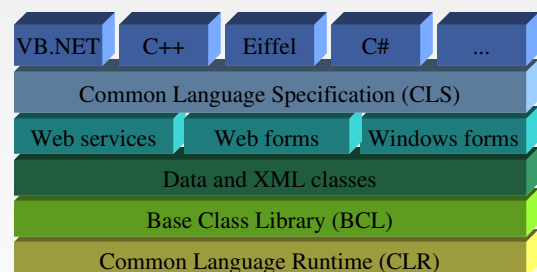
Intro .net Archi .net Langages CLR BCL Conclu

Plan du cours

- C'est quoi .NET ?
- L'architecture .NET
- Les langages .NET
- Le Common Language Runtime (CLR)
- Les bibliothèques de base
- Les outils

Intro .net Archi .net Langages CLR BCL Conclu

L'architecture .NET 2.0



Intro .net Archi .net Langages CLR BCL Conclu

.NET : pour quelles applications ?

- Conçu pour faciliter le développement d'applications n-tiers
 - utilisation/création facile de services web
 - accès facile aux données (ADO.NET)
 - présentation par pages web dynamiques (ASP.NET)

Intro .net Archi .net Langages CLR BCL Conclu

Les langages .NET

- Le langage "dédié" : C#
 - proche de java
- Les classiques
 - VB.NET, C++, J#, Eiffel#
- Mais aussi :
 - ADA, Scheme, Python, Fortran, ...
- En tout une vingtaine de langages supportés

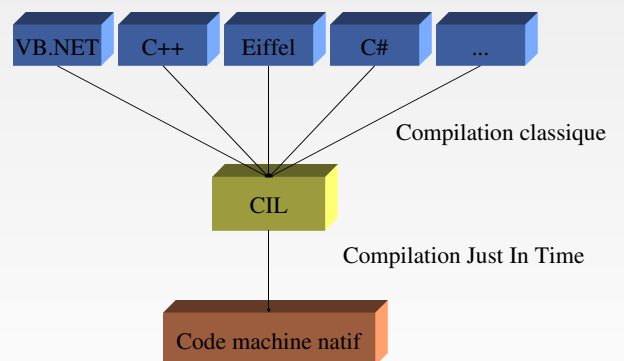
Intro .net Archi .net Langages CLR BCL Conclu

Langage .NET et MSIL

- Un langage .NET
 - est compilé vers du MSIL
 - est conforme à la CLS
- MSIL = CIL
 - = Microsoft Intermediate Language
 - = Common Intermediate Language
 - ≈ Byte code Java
 - interprété par un moteur d'exécution (machine virtuelle)

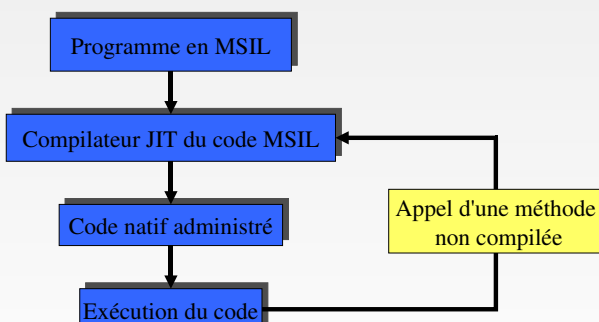
Intro .net Archi .net Langages CLR BCL Conclu

Compilation



Intro .net Archi .net Langages CLR BCL Conclu

Compilation Just In Time



Intro .net Archi .net Langages CLR BCL Conclu

Common Language Specification

- Contient la spécification des langages .net
- Entre autres :
 - le système de types
 - un ensemble de règles
 - règle 7 : "The underlying type of an enum shall be a built-in CLS integer type."
 - règle 25 : "The accessibility of a property and of its accessors shall be identical."
 - ...

Intro .net Archi .net Langages CLR BCL Conclu

Managed / unmanaged code

- Managed code
 - code écrit pour le CLR
 - code "géré" par le CLR : ramasse-miettes, politique de sécurité, vérification du typage à l'exécution, ...
- Unmanaged code
 - code non géré par le CLR
 - exécution plus rapide

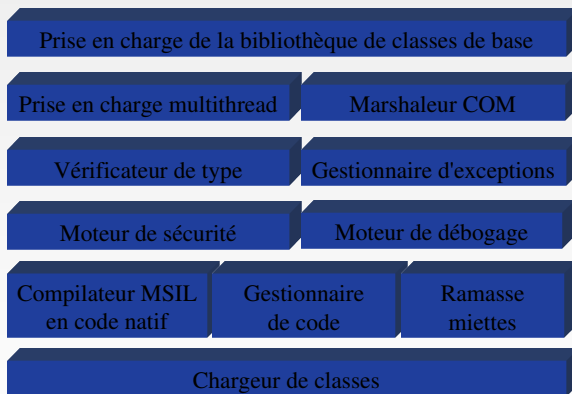
Intro .net Archi .net Langages CLR BCL Conclu

Le statut à part de C++ :

- Il y a le managed C++ et le unmanaged C++ ...
- Le managed C++ est une extension au C++ pour satisfaire les CLS
- Cohabitation possible entre du C++ managed et du C++ unmanaged (spécificité C++)
 - réutilisation de C++ existant
 - performances
- Le managed C++ a évolué en C++/CLI en 2005
 - syntaxe + agréable, meilleure intégration au CLI (Common Language Infrastructure), + proche de C++

Intro .net Archi .net Langages CLR BCL Conclu

Common Language Runtime



Intro .net Archi .net Langages CLR BCL Conclu

Composants principaux du CLR

- Système de type
- Système de métadonnées
- Système d'exécution

Intro .net Archi .net Langages CLR BCL Conclu

Le système de types

- 2 sous-systèmes :
 - types "valeur" (value types)
 - de base ou construits par le programmeur
 - types "référence" (reference type)
 - objet
 - Tous les objets héritent de la classe object
 - Méthodes : equals, finalize, GetHashCode, getType, memberwiseClone, ToString
 - interfaces
 - pointeurs

Intro .net Archi .net Langages CLR BCL Conclu

Built-In Types

- Bool
- Char (16-bit unicode)
- Entiers signés et non signés, sur 8, 16, 32 et 64 bits
- Flottants sur 32 et 64 bits
- Object
- String
- Flottants et entiers dépendant de la machine

Intro .net Archi .net Langages CLR BCL Conclu

Les métadonnées

- Pour stocker des informations sur les composants
- Les compilateurs stockent des infos sur les types
 - pour l'interopérabilité
 - pour le moteur d'exécution (gestion mémoire par exemple)
- Extension des métadonnées possible

Intro .net Archi .net Langages CLR BCL Conclu

Assemblies et Manifests

- Assembly
 - un manifeste
 - des classes de n'importe quel langage placées dans un espace de nom
 - des méta-données (images, textes, vidéos, ...)
 - créé par al.exe
- Manifeste
 - informations de nom et de version
 - liste des types et l'endroit où ils sont dans l'assembly
 - informations de dépendance

Intro .net Archi .net Langages CLR BCL Conclu

Système d'exécution

- Chargement et vérification de code
- Gestion des exceptions
- Compilation JIT
- Gestion mémoire
- Sécurité

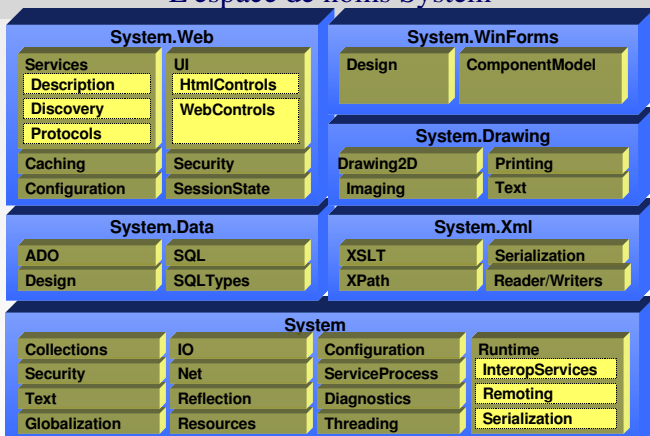
Intro .net Archi .net Langages CLR BCL Conclu

La bibliothèque de classes

- Commune pour tous les langages
- Microsoft.Csharp, VisualBasic, JScript
 - classes pour supporter la compilation et la génération de code dans différents langages
- Microsoft.Vsa
 - pour l'intégration de scripts dans les applications
- Microsoft.Win32
 - classes pour gérer les exceptions systèmes et pour manipuler les registres du système
- System

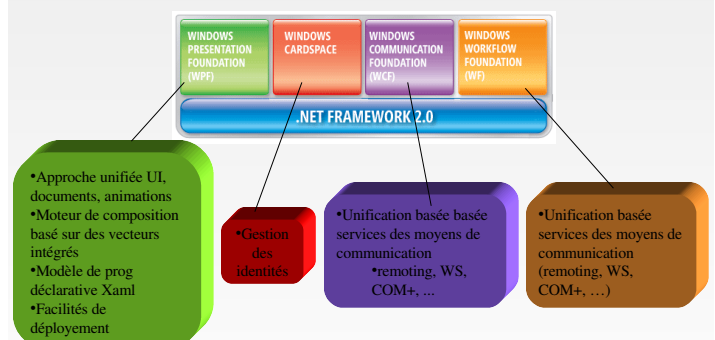
Intro .net Archi .net Langages CLR BCL Conclu

L'espace de noms System



Intro .net Archi .net Langages CLR BCL Conclu

.net 3.0 (winFX)



Intro .net Archi .net Langages CLR BCL Conclu

Conclusion

- .NET et J2EE
 - .NET+COM+ \cong J2SE + J2EE
 - efficacité ?
- Plusieurs langages et un langage dédié (C#)
- Plateforme cible principale : Win
- Une volonté de Microsoft vers des solutions non propriétaires
 - standardisation ECMA/ISO de C#
 - implémentations open source

Intro .net Archi .net Langages CLR BCL **Conclu**

Les outils

- CLR
 - .net framework (gratuit !) - SDK, runtime
 - Rotor (win, FreeBSD, MacOS X)
 - Mono (Linux)
 - DotGNU (Linux)
- IDE
 - VisualStudio (payant ...)
 - SharpDevelop (gratuit)
 - WebMatrix (gratuit)
 - ...

Intro .net Archi .net Langages CLR BCL **Conclu**