



# Tema 2: Introducción a la plataforma Microsoft .NET

# Índice

- Elementos de la plataforma .NET
- .NET Framework y sus componentes
- Visual Studio .NET y sus elementos
- Proceso de desarrollo de una aplicación

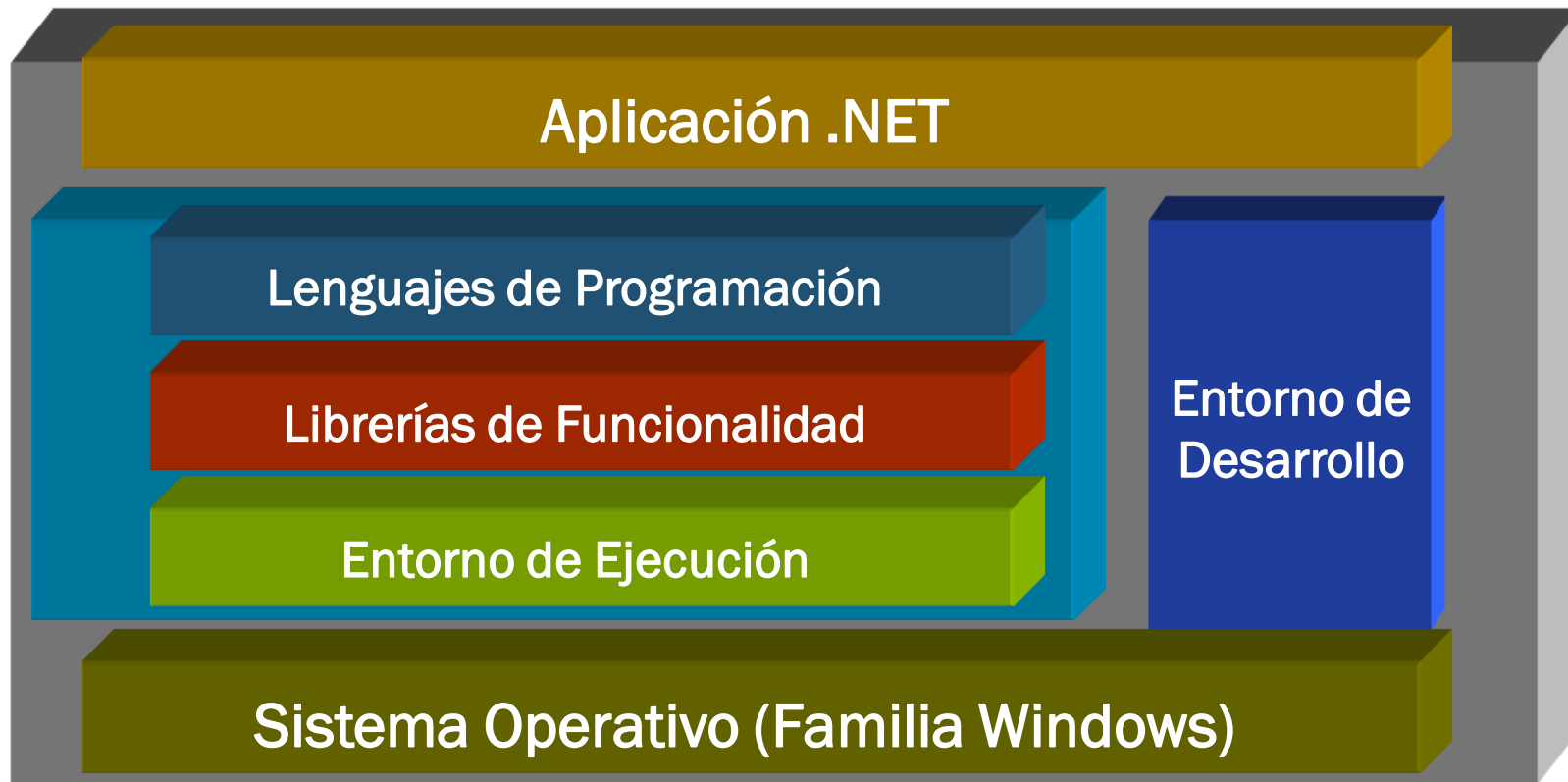
# ¿Qué es Microsoft .NET?

- Plataforma de Desarrollo compuesta de:
  - Entorno de Ejecución (Runtime)
  - Bibliotecas de Funcionalidad (Class Library)
  - Lenguajes de Programación
  - Compiladores
  - Herramientas de Desarrollo (IDE & Tools)
  - Guías de Arquitectura

# Características de Microsoft .NET

- Plataforma de ejecución intermedia
- 100% Orientada a Objetos
- Multilenguaje
- Modelo de Programación único para todo tipo de aplicaciones y dispositivos de hardware
- Se integra fácilmente con aplicaciones existentes desarrolladas en plataformas Microsoft
- Se integra fácilmente con aplicaciones desarrolladas en otras plataformas

# Plataforma de Ejecución Intermedia



# ¿Qué es el .NET Framework?

- Paquete de software fundamental de la plataforma .NET. Incluye:
  - Entorno de Ejecución (Runtime)
  - Bibliotecas de Funcionalidad (Class Library)
- Se distribuye en forma libre y gratuita
- Existen tres variantes principales:
  - .NET Framework Redistributable Package
  - .NET Framework SDK
  - .NET Compact Framework

# Variantes de .NET Framework

- **.NET Framework Redistributable Package**
  - Este es el mínimo componente de la plataforma .NET que se necesita para poder ejecutar aplicaciones.
  - Está compuesto por:
    - El entorno de ejecución de la plataforma .NET
    - Las bibliotecas de funcionalidad reutilizable

# Variantes de .NET Framework

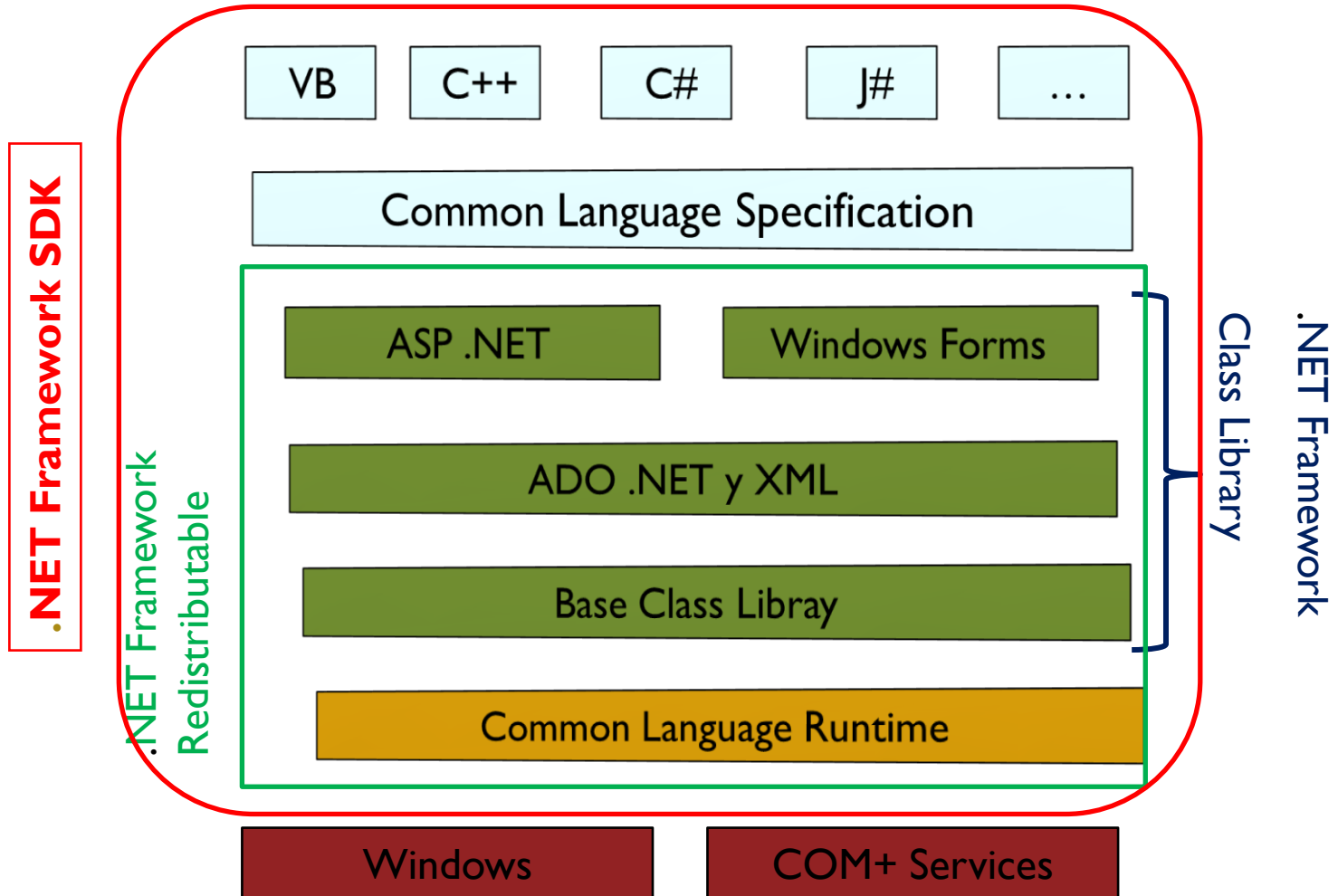
- **.NET Framework SDK:**
  - Esta versión contiene herramientas de desarrollo de línea de comandos (compiladores, depuradores, etc.), documentación de referencia, ejemplos y manuales para desarrolladores de aplicaciones.
  - Suele ser la variante que se instala en los entornos de desarrollo de aplicaciones, y es más útil a los programadores que a los usuarios finales.
  - Para poder instalar la versión SDK (Software Development Kit) es necesario instalar previamente el Redistributable Package.
- **.NET Compact Framework**
  - Esta es una versión reducida del .NET Framework Redistributable, especialmente pensada para ser instalada en dispositivos móviles como Pocket PC's y SmartPhones.



# ¿Dónde instalar .NET Framework?

|  | Cliente                | Servidor |
|--|------------------------|----------|
| Aplicación de Escritorio               | ✓                      | ✓ *      |
| Aplicación Web                         |                        | ✓        |
| Aplicación de Consola                  | ✓                      | ✓ *      |
| Aplicación Móvil                       | .NET Compact Framework |          |
| * Sólo si la aplicación es distribuida |                        |          |

# Arquitectura de .NET Framework



# CLR: Common Language Runtime

- El CLR es el motor de ejecución (runtime) de .NET
- Características
  - Compilación Just-In-Time (JIT)
  - Gestión automática de memoria (Garbage Collector)
  - Gestión de errores consistente (Excepciones)
  - Ejecución basada en componentes (Assemblies)
  - Gestión de Seguridad
  - Multithreading

# .NET Framework Class Library

- Conjunto de Tipos básicos (clases, interfaces, etc.) que vienen incluidos en el .NET Framework
- Los tipos están organizados en jerarquías lógicas de nombres, denominados NAMESPACES
- Los tipos son INDEPENDIENTES del lenguaje de desarrollo
- Es extensible y totalmente orientada a objetos

# .NET Framework Class Library

SYSTEM (Namespace Raíz)

## System.Web

Services

Description

Discovery

Protocols

Caching

Configuration

UI

HtmlControls

WebControls

Security

SessionState

## System.Windows.Forms

Design

ComponentModel

## System.Drawing

Drawing2D

Imaging

Printing

Text

## System.Data

OleDb

Common

Odbc

SqlClient

## System.Xml

XSLT

XPath

Serialization

## System

Collections

Configuration

Diagnostics

Globalization

IO

Net

Reflection

Resources

Security

ServiceProcess

Text

Threading

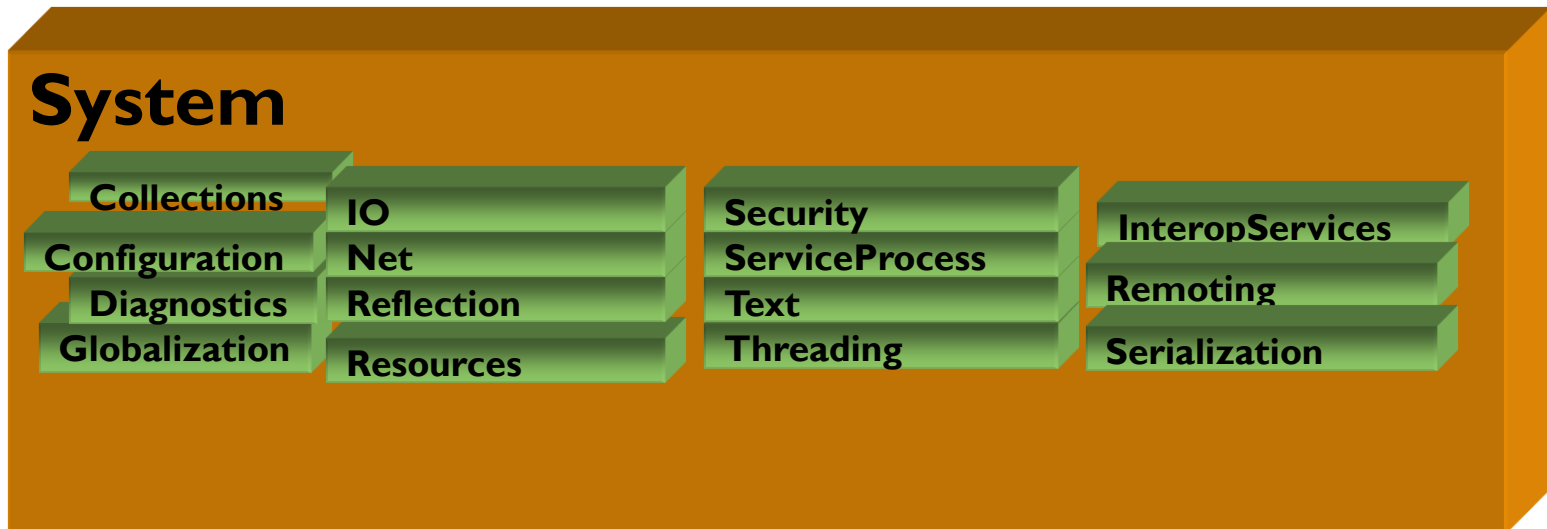
Runtime

InteropServices

Remoting

Serialization

# Base Class Library



# Acceso a Datos: ADO.NET

## System.Data

**Common**

**OracleClient**

**Odbc**

**SqlClient**

**OleDb**

**SqlTypes**

## System.Xml

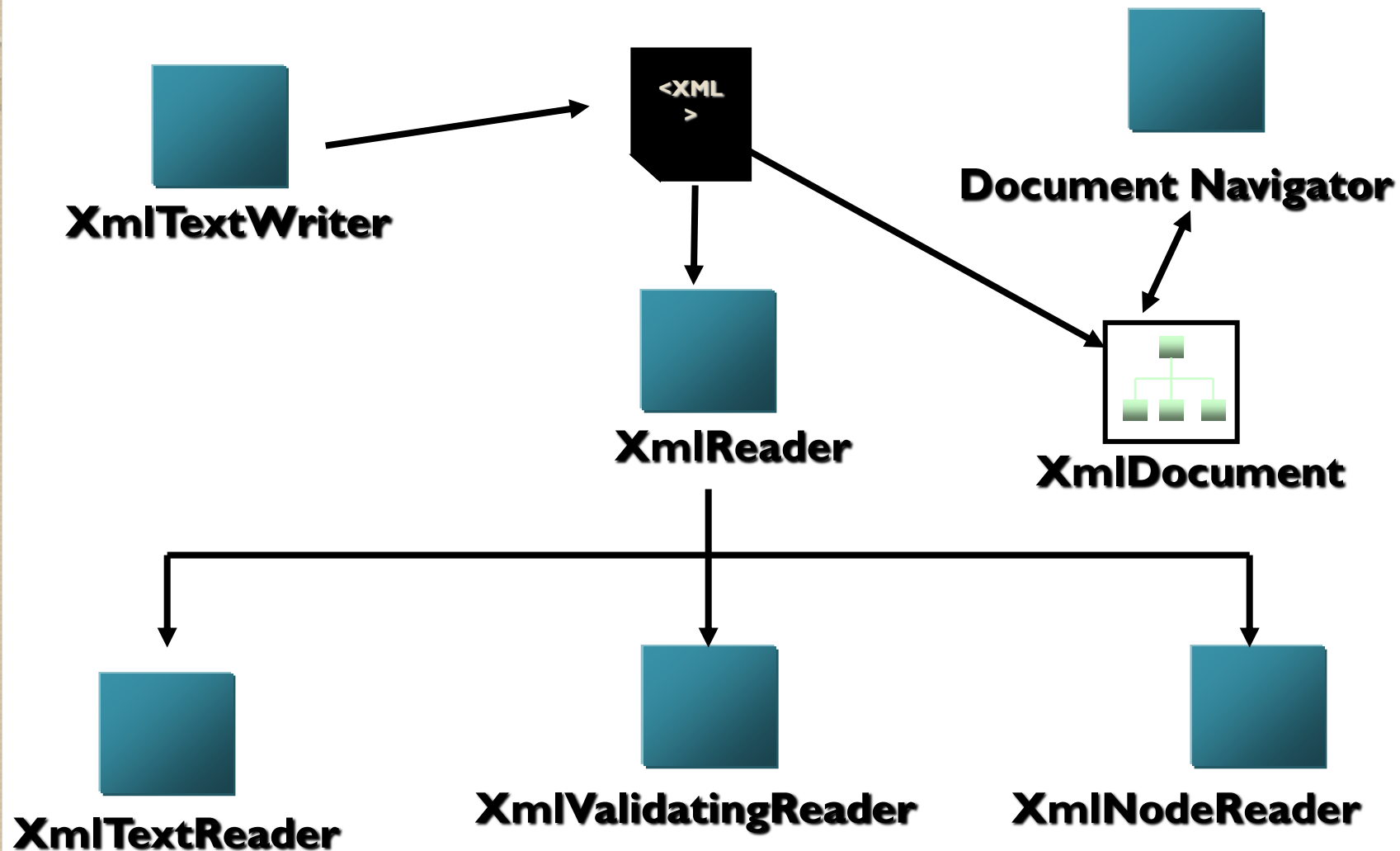
**XSLT**

**XPath**

**Serialization**

**Schema**

# ADO .NET – Soporte XML





# Windows Forms

## **System.WinForms**

**Design**

**ComponentModel**

## **System.Drawing**

**Drawing2D**

**Printing**

**Imaging**

**Text**

# Aplicaciones Web:ASP .NET

## System.Web

### Services

Description

Discovery

Protocols

### UI

HtmlControls

WebControls

Caching

Security

Configuration

SessionState

# Common Language Specification

- Especificación que estandariza una serie de características soportadas por el CLR
- Contrato entre diseñadores de lenguajes de programación y autores de bibliotecas
- Permite la interoperabilidad entre lenguajes
- Microsoft provee implementaciones de 4 lenguajes, todos compatibles con CLS
  - Microsoft Visual Basic .NET
  - Microsoft Visual C# .NET
  - Microsoft Visual J#.NET
  - Microsoft Visual C++.NET

# CLS – Elección del Lenguaje

- .NET posee un único runtime (el CLR) y un único conjunto de bibliotecas para todos los lenguajes
- No hay diferencias notorias de performance entre los lenguajes provistos por Microsoft
- El lenguaje a utilizar, en gral., dependerá de su experiencia previa con otros lenguajes o de gustos personales
  - Si conoce Java, Delphi, C++, etc. → C#
  - Si conoce Visual Basic o VBScript → VB.NET
- Los tipos de aplicaciones .NET son **INDEPENDIENTES** del lenguaje que elija