

## Introdução: A Plataforma .NET

- **Infraestrutura para a Plataforma .NET**
  - Uma colecção de classes
    - System, System.Data, System.Web, System.Windows, etc.
    - Performance Monitor, Event Log, XML Web Services, Web Forms, etc.
  - Runtime for applications
    - Common Language Runtime
    - Common type specification for languages

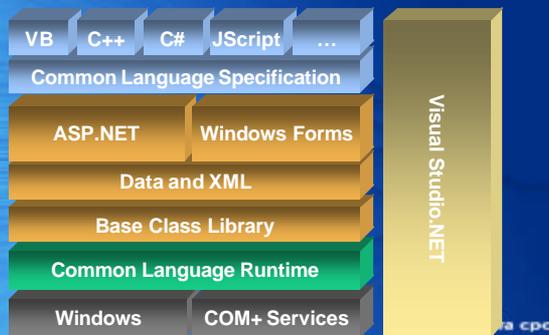


## Introdução: Web Standards/Práticas

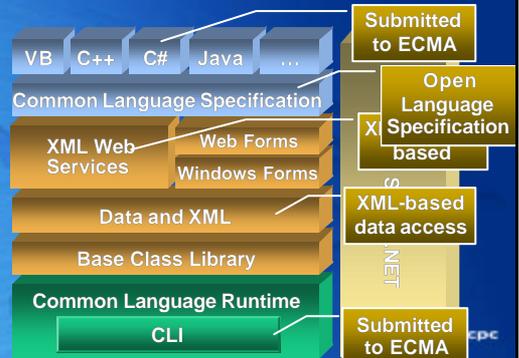
- A plataforma .NET suporta
  - HTML, XML, WSDL, SOAP, XSLT, XPath ...
- **Aplicações distribuídas Internet-ready**
  - Modelo tradicional, fortemente ligado e com estado não funciona na Web
  - A plataforma.NET favorece XML Web Services fracamente ligados, distribuídos sem estado
- **Aplicações Windows**

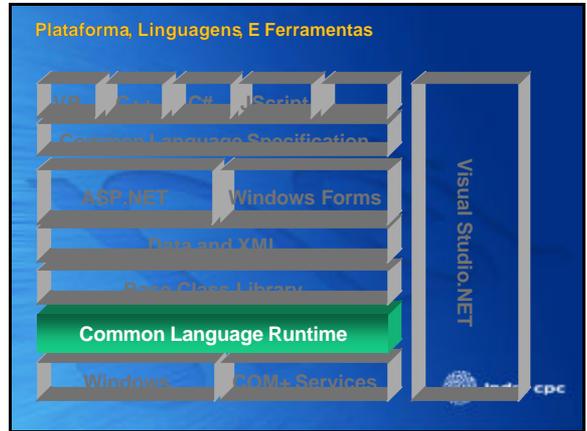
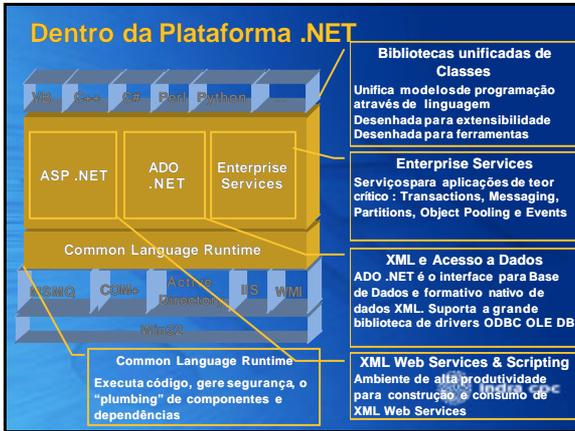


## Plataforma, Linguagens, E Ferramentas

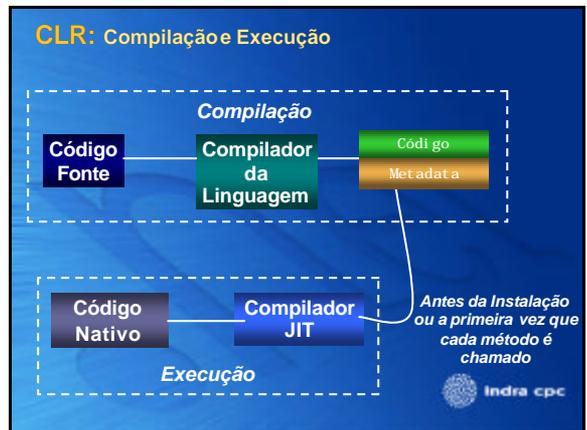


## Introdução: .NET Open Platform e Ferramentas

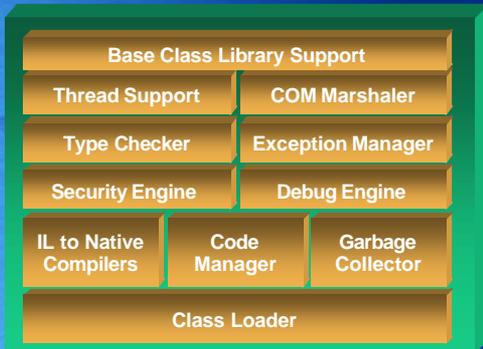




- ### Common Language Runtime: Características
- **Diversos modelos de compilação**
    - Nativo (ex: Managed C++)
    - Managed
      - MSIL (ex: VB, C#, C++)
      - MSIL é compilado para nativo antes da execução
    - Sem interpretador: IL em instalação ou execução para compilação para código nativo
  - **Correção do código e segurança em tipos**
    - IL pode ser verificado para garantir a segurança em tipos
    - Não existência de: Casts inseguros variáveis não inicializadas e indexação fora dos limites de um vector
- Indra cpc



## CLR: Componentes



cpc

## CLR: Desenvolvimento Simplificado

- Elimina COM “plumbing”
- Não existência de
  - Manipulação do Registry => componentes auto-descritivos
  - GUID's => namespaces hierárquicos
  - Ficheiros .IDL => código fonte a metadata
  - HRESULT's => exceções estruturadas
  - IUnknown => class de topo: OBJECT
  - AddRef/Release => garbage collector
  - CoCreateInstance => operador “new”

Indra cpc

## Desenvolvimento Multi-Linguagem

- Object system é embebido, não montado
  - Escolhan a linguagem preferida
  - Não é necessário aprender novas regras e APIs
- Características avançadas de multi-linguagem
  - Herança entre linguagens e exceções
- Suporta mais de 20 linguagens
- Ferramentas de suporte de alto nível
  - Debuggers, etc ... funcionam com todas as linguagens

Indra cpc

## Desenvolvimento Simplificado

- Não é necessário manipulação do Registry
  - Código é auto-descritivo
  - Os componentes são copiados para a directoria da aplicação
- Instalação Segura
  - A instalação de uma aplicação não afecta outra existente
- Execução side-by-side
  - Várias versões do mesmo componente podem coexistir

Indra cpc

## Desenvolvimento Simplificado

- **Assemblies .NET**
  - A unidade de distribuição de manutenção de versões e segurança
  - Similar a DLL's, mas auto-descritivas através de um manifest (ficheiro)
- **.NET suporta remoting dos seus componentes**
  - Similar a DCOM
  - Utiliza
    - HTTP ou TCP/IP Sockets
    - Necessita de alterações ligeiras aos componentes



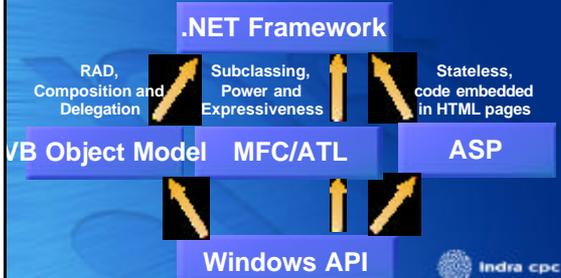
## Integração Simplificada

- **Uma classe .NET pode ser um objecto COM**
  - Não é necessário desenvolvimento extra
- **Objectos COM podem ser importados**
  - Não são necessárias COM API's
- **Classes .Net usam serviços COM+**
  - Transactions, Object pooling, etc...
- **Chamadas a funções de sistema**
  - Para os casos mais comuns não é necessário marshaling

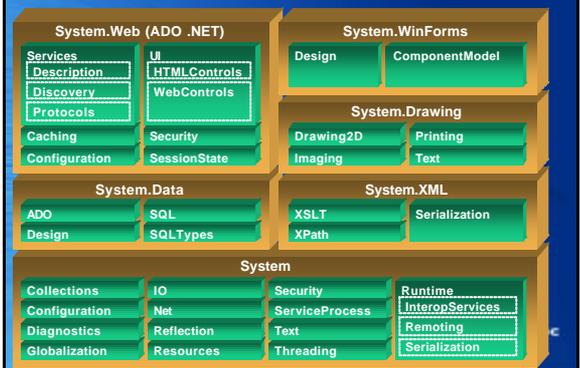


## Um modelo de programação unificado

Consistent API availability regardless of language and programming model



## Classes unificadas



## Utilização Simples

- **Organização**
  - Organização em namespaces hierárquicos
- **Sistema de tipos unificado**
  - Tudo é um objecto
  - Boxing converte value types para objectos
    - Não há necessidade do tipo especial Variant
- **Orientado ao Componente**
  - Propriedades, métodos e eventos são primeira classe
  - Funcionalidade design-time



## Mais simples. Como?

C++

### Windows API (C++)

```
HWND hwndMain = CreateWindowEx (
    0, "MainWClass", "Main Window",
    WS_OVERLAPPEDWINDOW | WS_HSCROLL | WS_VSCROLL,
    CW_USEDEFAULT, CW_USEDEFAULT,
    CW_USEDEFAULT, CW_USEDEFAULT,
    (HWND) NULL, (HMENU) NULL, hInstance, NULL);
ShowWindow(hwndMain, SW_SHOWDEFAULT);
UpdateWindow(hwndMain);
```

### Plataforma .NET (C#)

```
Form form = new Form();
form.Text = "Main Window";
form.Show();
```



## Mais simples. Como?

Visual Basic 6

### Operadores unários e inicializações de variáveis

A atribuição e modificação dos valores das variáveis é mais simples

#### VB6:

```
Dim myString As String
myString = "Hello "
myString = myString & "World"
```

#### VB .NET:

```
Dim myString As String = "Hello "
myString &= "World"
```



## Mais simples. Como?

Visual Basic 6

### Tratamento de Erros - VB6

```
On Error Goto MyErrorTrap
```

```
....
```

```
MyErrorTrap:
    MsgBox Err.Description
```

Neste caso isolamos o tratamento do erro da lógica da aplicação. Bom para métodos pequenos mas um pesadelo para grandes métodos. No VB.NET pode-se continuar a usar o goto ou SEH (Structured Exception Handling) para o tratamento de erros. O SEH mantém o código de tratamento de erro com a lógica da aplicação.

#### VB .NET:

```
Try
....
Catch ex As Exception
    MsgBox ex.Message
End try
```



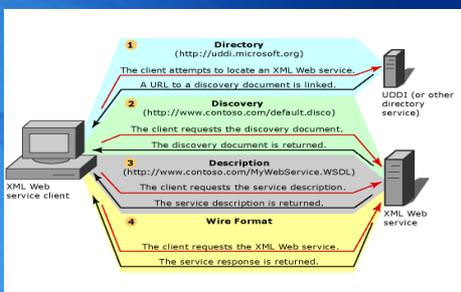
## Resumo

- .NET é baseado em standards
- Plataforma comum para todas as linguagens
  - Common Language Runtime
  - Suporte Multi-Linguagem
- Desenvolvimento e distribuição simplificados
  - Classes Unificadas
  - Funciona com COM
  - Assemblies

## Web Services - Introdução

- Responde a necessidades colocadas nas aplicações distribuídas
- Permite comunicação entre sistemas heterogêneos
- É a camada de lógica de negócio no modelo de três camadas (three-tier model)
- Acesso remoto a recursos de uma maneira fracamente ligada
- Assunção de que o cliente e o Web Service entendem as mensagens trocadas
- Utiliza o protocolo SOAP para a transferência de mensagens XML

## Web Services - Infraestrutura



## Recursos

- *Frequently asked questions about Microsoft .NET Framework* - Esta página contém grande parte das perguntas que mais frequentemente são colocadas em relação à plataforma .NET. <http://msdn.microsoft.com/library/techart/faq111700.htm>
- *The GotDotNet Home Page* - Um excelente site para se encontrar mais informação e alguns exemplos de aplicações reais .NET. <http://www.getdotnet.com/>
- *An Introduction to the .NET Framework* - Uma introdução simples e concisa à plataforma .NET.
- [http://www.getdotnet.com/about\\_new.aspx](http://www.getdotnet.com/about_new.aspx)
- *Your first .Net Application* - Um filme que mostra um exemplo de uma aplicação ASP.NET que consome Web Services. <http://msdn.microsoft.com/theshow/Episode013/>
- *Using Soap* - Um filme sobre a utilização de SOAP e da sua relação com Web Services. <http://msdn.microsoft.com/theshow/Episode010/default.asp>