

# Ignify Consulting

## C #

Author: Seema Gopalan  
seema@ignify.com

Ignify Consulting  
13304 Alondra Blvd #201.  
Cerritos CA 90703  
[www.ignify.com](http://www.ignify.com)

March 2003



**Confidential**

Property of Ignify, Copyright 1999 - 2003  
May not be reproduced or distributed without prior permission

1

## Topics covered:

- Introduction to .NET framework
- Features in c#
- Features inherited from other languages
- What 's new in c#



**Confidential**

Property of Ignify, Copyright 1999 - 2003  
May not be reproduced or distributed without prior permission

2

## **.NET Runtime features :**

- Garbage Collection
- Exception handling
- Events
- Language Independent
- Runtime type verification
- Meta Data

## **.NET Framework**

Web Services

Web Forms

Windows Forms

Data & XML Classes

ADO.Net, SQL, XSLT, XPATH, XML etc

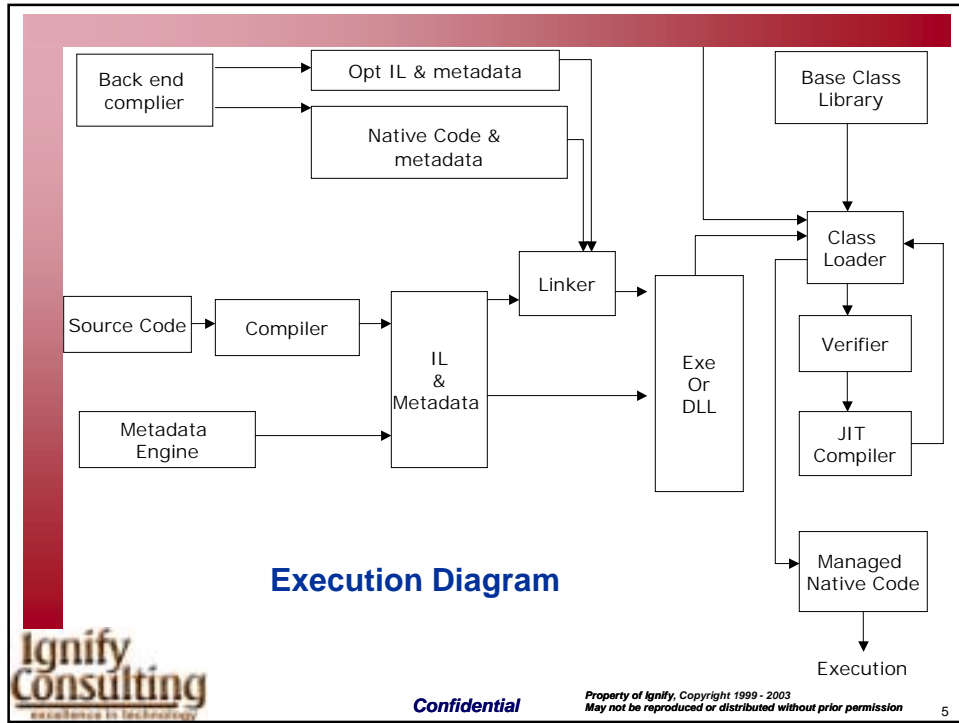
Frame Work Base Classes

IO,string,.net,security,threading,reflection,Collection

Common Language Runtime

Debug, exception, type check, JIT Compiler

Windows Platform



## Features of C# :

<ul style="list-style-type: none"> <li><b>Static Typing</b></li> <li><b>Runtime Polymorphism</b></li> <li><b>exceptions</b></li> <li><b>Reference type</b></li> </ul> <p style="text-align: center;">C#, C++ and Java</p> <ul style="list-style-type: none"> <li><b>compile time polymorphism</b></li> <li><b>value types</b></li> <li><b>operator overloading</b></li> </ul> <p style="text-align: center;">C# and c++</p> <ul style="list-style-type: none"> <li><b>unified type system</b></li> </ul>	<ul style="list-style-type: none"> <li><b>virtual execution system</b></li> <li><b>Object super base class</b></li> <li><b>single public inheritance</b></li> <li><b>multiple public realization</b></li> <li><b>non deterministic finalization</b></li> <li><b>garbage collection</b></li> <li><b>multi threading</b></li> <li><b>reflection</b></li> </ul> <p style="text-align: center;">C# and Java</p>
--	---

**Ignify Consulting**  
excellence in technology

**Confidential**

Property of Ignify, Copyright 1999 - 2003  
May not be reproduced or distributed without prior permission

6

## Sample code of IL or MSIL

```
HikerMain: void()
.method public hidebysig static void Main() il managed
{
    .entrypoint
    // Code size      23 (0x17)
    .maxstack 2
    .locals (int32 V_0,
            int32 V_1)
    IL_0000: ldc.i4.s 54
    IL_0002: stloc.0
    IL_0003: ldc.i4.s 13
    IL_0005: stloc.1
    IL_0006: ldloc.0
    IL_0007: ldloc.1
    IL_0008: div
    IL_0009: call      void [mscorlib]System.Console::WriteLine(int32)
    IL_000e: ldloc.0
    IL_000f: ldloc.1
    IL_0010: rem
    IL_0011: call      void [mscorlib]System.Console::WriteLine(int32)
    IL_0016: ret
} // end of method Hiker::Main
```

## Sample Code Of C#

```
using System;
public class HelloWorld {
    public static void Main() {
        // Use the system console object
        System.console.WriteLine("Hello World");
    }
}
```

## C++ Preprocessor Holdovers

- #ifdef - #define - #ifndef - #endif
  - compile direction only, no macro direction
- No actual preprocessor
  - no # include, no #pragma

## Visibility

- public, private, protected, internal
- Internal is assembly scope
  - Basic compiled unit

## Static & Const

- Static
- Const
- Read Only
  - For classes and structs that need to be constructed, but are constant

## NonJava Non C++ Keywords

- Foreach (int a in intArray)
  - iterate through an array or indexable type
- Lock
  - Multithreaded exclusion
- Checked or unchecked
  - Numerical overflow

## Function Side Effects

Void int AnyFunction{ret int x, out int y, int z }

- - ret
  - Return types, allows pass by reference
    - must be initialized before call
- - out
  - input/ output values

## Class Casting

Typeof (java instanceof)

- for class casting

as

- forces a cast to an interface

is

- checks if an object implements an interface
- if (obj is IMyInterface)

## Properties: Built-in Accessors

- `get()`
  - Automatically invoked when an instance is on the right of an assignment
- `set()`
  - Automatically invoked when an instance is on the left of an assignment

```
Public int s {  
    get{return s;}  
    set{s=value;} //value is a keyword,of type s  
};  
s=3; // Implicit invocation of s.set()
```

## Value & Reference Type

- **Class**
  - Reference type
  - Heap Allocated
- **Struct**
  - Value type
  - Stack allocated
- **Interface**
  - For multiple inheritance

## Versioning & Virtual Functions

- **new**
  - operator applied to subclass functions indicating they are a “new” name than an inherited virtual function
- **Override**
  - operator applied to subclass functions indicating they override a similarly named virtual function
  - class c { virtual foo(); };  
class d : c { override foo(); new foo(); };

## Operator Overloading

Overload any C++ operator

overloaded [x] operator is special: an “indexer”

```
class c {  
    public object this.[int i] {  
        get{return x[i];}  
        set{x[i] = value;}  
    };  
};
```

## Unsafe Code

- Unsafe
  - allows pointer arithmetic
- Fixed
  - specifies the location in memory must stay the same

### Code using unsafe

- fails type checking
- cannot be downloaded and run on foreign system

## Attributes

- Compiler generated or User defined
- live in the program meta data
- dynamically query able
- inserted in code like comments

## XML + Attributes = JavaDoc

Parsed by the parser, rather than a separate engine  
XML comments describe code

- Comments type checked with code they reference
- Generates XSL documentation page

- **Los Angeles Office**  
13304 Alondra Blvd #201, Cerritos CA 90703  
Email: sales@Ignify.com  
Tel: 562-404-8089
- **India Office**  
7 Madhuban, North Main Road, Koregaon Park, Pune  
Tel: +91-20-612-0778  
Email: India@Ignify.com
- **San Francisco Bay Area**  
4800 Great America Pkwy, Suite 310, Santa Clara, CA 95054  
Tel: 408-480-3289
- **Global Website:** <http://www.Ignify.com>

