

.NET'İN TARİHSEL GELİŞİM SÜRECİ

Net ortamının ve C# programlama dilinin Microsoft tarafından ilk duyurusu 2000 yılında yapılmıştır. .Net ortamını kullanan ilk resmi IDE “Visual Studio 2000” ile 2002 senesinde çıkmıştır. Bunu Visual Studio 2003 ve Visual Studio 2005 versiyonu izlemiştir.

C# programlama dili “**Anders Hejlsberg**” tarafından tasarlanmıştır. C# büyük ölçüde Java’dan alıntı yapılarak tasarlanmıştır. Resmi tasarım ekibi, “Anders Hejlsberg” in de içinde bulunduğu 4 kişiden oluşmaktadır. Anders Hejlsberg, eski bir Borland çalışanıdır ve “Borland Turbo Pascal” ve “Delphi” ürünlerinin tasarımcılarından.

Microsoft, C# dilinin ve .NET ortamının yaygınlaşması için bunları tekelden çıkartmak istemiştir. C# programlama dili ECMA (European Computer Manufacturing Association / www.ecma-international.org) tarafından 2003 yılında “C# Language Specification Second Edition” ismiyle standardize edilmiştir. Bu standardın ECMA kod adı “ECMA-334” tür. Web sayfasından indirilebilir. Bir yıl sonra C# ISO tarafından da ISO/IEC 23270:2003 kod adı ile standardize edilmiştir. Bu standardın ECMA 334 ten bölüm numaralandırmaları dışında içerik bakımından bir fark yoktur.

ECMA ayrıca .NET ortamındaki programlama dillerinin sağlaması gereken ortak özellikleri *CLI (Common Language Infrastructure)* ismiyle ECMA-335 kod numarasıyla standardize etmiştir. Bu standardın ISO karşılığı ISO/IEC:23271:2003 dir.

Daha sonra C# ‘a bazı özellikler eklemiştir. ECMA uyarı eklentileri, “ECMA-334 4th Edition/June 2006” başlığı altında standardize etmiştir. Bu yeniliklerden CLI standartları da etkilenmiştir ve “ECMA-335 4th Edition” versiyonu da oluşturulmuştur

Ecma standartlarının web adresleri :

- <http://www.ecma-international.org/publications/standards/Ecma-334.htm>
- <http://www.ecma-international.org/publications/standards/ECMA-335.HTM>

ISO standartlarının web adresi :

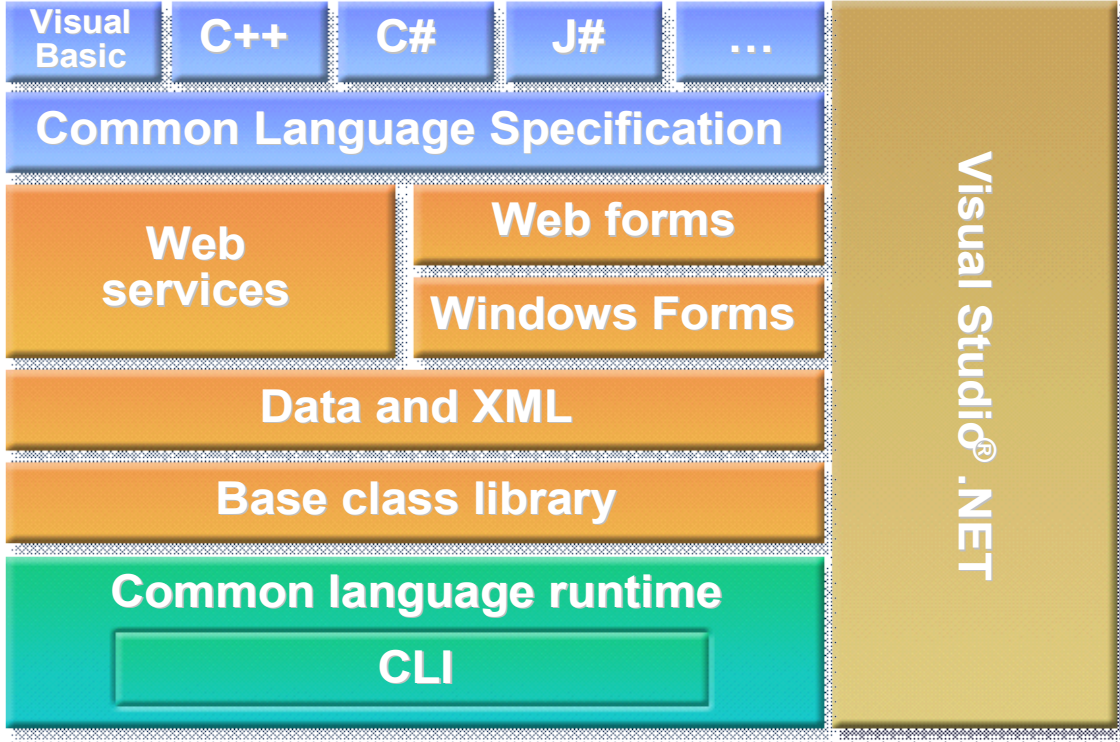
- http://isotc.iso.org/livelink/livelink/fetch/2006/2489/Ittf_Home/PubliclyAvailableStandards.htm

ISO standart numaraları :

- ISO/IEC 23270:2006
- ISO/IEC 23271:2006
- ISO/IEC 23272:2006

C# ve .NET ÇALIŞTIRMA ORTAMLARININ ELDE EDİLMESİ

.Net çalıştırma ortamı ve C# dilinin komut satırlı derleyicisi bedava dağıtılmaktadır. Yani komut satırı derleyicisi kullanarak C# ta program yazmak için mutlak anlamda bir ürün satın almaya gerek yoktur.



.NET Framework

.Net çalışma ortamının sağlanması için sunulan iki çeşit paket vardır:

- .NET Framework Version 2.0 **Redistributable** Package
- .NET Framework Version 2.0 **Software Development Kit**

.NET framework, kendi içerisinde C# derleyicisini ve debug programlarını da içermektedir. .NET SDK içerisinde geliştiricilere özel dokümantasyon ve örnek kodlar da mevcuttur. Paketlerin aşağıdaki web adreslerinden indirilebilir :

- <http://www.microsoft.com/downloads/details.aspx?displaylang=tr&FamilyID=0856eacb-4362-4b0d-8edd-aab15c5e04f5>

- <http://www.microsoft.com/downloads/details.aspx?familyid=fe6f2099-b7b4-4f47-a244-c96d69c35dec&displaylang=en>

C# ve .NET ÇALIŞTIRMA ORTAMLARININ ELDE EDİLMESİ

C# dili içinde komut satırlı derleyicinin yanında kısıtlı bir IDE olan Visual Studio C# Express Edition ve MSDN Express Edition ücretsiz olarak web sitesinden indirilebilir:

- <http://msdn.microsoft.com/vstudio/express/visualcsharp/download/>

İşletim sistemleri Windows 2000 den bu yana .NET çalışma ortamlarını zaten içerisinde barındırmaktadır. Windows 95,98 ve ME gibi eski sistemlerde bu çalıştırma ortamının “.NET FRAMEWORK” ismiyle kurulması gerekir

Linux işletim sisteminde .NET çalışma ortamı sağlanması için MONO isimli bir proje yürütülmektedir.

- http://www.mono-project.com/Main_Page

Bu proje tamamen GNU lisanslıdır. (GNU :General Public License - Free Software Foundation (FSF) tarafından geliştirilmiştir.)

FreeBSD ve Mac OS X işletim sistemlerinde .NET çalışma ortamı sağlanması için ROTOR isimli bir proje yürütülmektedir.

- <http://research.microsoft.com/programs/europe/rotor/2004Projects.aspx>
- http://www.southern-storm.com.au/portable_net.html

.NET ÇALIŞTIRMA ORTAMI NEDİR?

.NET çalıştırma ortamı, Java gibi, “ara kod tabanlı” bir ortamdır. Yani bu ortamda programlar gerçek makine kodlarını değil standart bir “ara kod” içermektedir. Bu ara kod yorumlanarak (interpreter) çalıştırılmaktadır.

.NET ortamında CLI uyumlu diller ortak bir ara kod üretmektedir. Bu ara kod *IL (intermediate Language)* dilidir. Böylece .NET ortamında projenin bir kısmı bir dilde diğer kısmı başka bir dilde yazılabilmektedir.

Bir programlama dilinde yazılmış olan bir sınıfın başka bir programlama dilinden kullanılabilmesi için *COM (Component Object Module)* ve türevleri olan sistemler geliştirmiştir. Bir sınıf COM belirlemelerine uygun olarak yazılırsa, işletim sisteminin desteğiyle, böyle bir kullanım mümkün olabilmektedir. COM modül oluşturmak hem zordur hem de bunun da aşılamayan problemleri vardır.

.NET ortamının en önemli yeniliklerinden biri diller arasındaki kolay entegrasyondur. CLI uyumlu bütün diller, ortak dillerin ve kütüphanelerin kullanıldığı ortak bir ara koda dönüşüm yapmaktadır. Böylece bir programlama dilinde yazılıp ara koda dönüştürülmüş olan bir programın artık hangi dilde yazıldığına bir önemi kalmamaktadır. *Java* dilinde de bir ara kod üretimi vardır. Microsoft, büyük ölçüde *java* sisteminden hareketle .NET’i geliştirmiştir fakat daha ileriye giderek diller arası entegrasyonu da sağlamıştır. .Net çalıştırma ortamında ortak bir sınıf kütüphanesi kullanılmaktadır.Örneğin; Visual Basic programcısının kullandığı kütüphane ile C# programcısının kullandığı aynıdır.

Ara kodlu çalışma sisteminin, resmi kaynaklara göre, ortalama %18 lik bir performans kaybı vardır. Fakat .NET sistemi zaten sistem programlama amaçlı değil, uygulama programcılığı amacıyla kullanıldığı için bu kaybın önemli olduğu söylenemez.

C# ve .NET ortamı, kritik sistem programlama için uygun değildir. Fakat bazı sistem programları yine de bu ortamda hızlı bir biçimde yazılabilir.

.NET çalıştırma ortamı, özellikle hızlı yazılım geliştirme (rapid application development) gereksinimini karşılamaya yöneliktir. Yani bu ortamda pek çok şey hazır olarak bulunmaktadır. Programcı, program yazan kişi değil, neredeyse monte eden kişi haline gelmiştir.

.Net ortamı “bileşen tabanlı (component base)” çalışma ortamını desteklemektedir. Bileşen tabanlı programlamanın nesne yönelimli programlamanın bir üst aşaması olduğu söylenebilir. Bu geliştirme modelinde sınıflar biçiminde bir araya getirilmiş bileşenler programcı tarafından monte edilmektedir.

.NET ortamı, ayrıca *ASP.NET* isimli basit bir web tabanlı geliştirme sistemine de sahiptir. Yani .NET sistemiyle web üzerinde çalışan uygulamalar kolay bir biçimde yazılabilmektedir.

.NET çalıştırma ortamı, “Pocket PC (Windows CE)” isimli el bilgisayarlarında da hızlı yazılım geliştirmek için kullanılabilir. Pocket PC'nin .NET çalıştırma ortamını da içermektedir. El bilgisayarları için yazılmış olan bir .NET programı, hiç değişiklik yapılmadan, farklı mikroişlemcilerin bulunduğu pocket PC'lerde kullanılabilir. Tabii .NET in el bilgisayarları için kullanılan modeli daha düşük bir konfigürasyon içermektedir. Yani, bu mini sistemlerde sınıf bazı kütüphanesinin bazı bölümleri yoktur. Bu kütüphane “*Compact Framework*” diye isimlendirilmiştir.

C# NASIL BİR DİLDİR?

C#, basit, nesne yönelimli ve bileşen tabanlı bir dildir. Programlama dili olarak C# büyük ölçüde JAVA'dan klonlanmıştır. JAVA'nin iyileştirilmiş bir biçimi olduğu söylenebilir. Fakat son yıllarda programlama dili olarak JAVA'da da geliştirmeler de yapılmıştır.

C# ve JAVA'nin kökeni C++'dir.C#, JAVA'dan daha fazla C++'a yakınlaştırılmıştır. Gerçekten de C++'da olan bazı özellikler C#'da olduğu halde JAVA'da yoktur. Aslında bu özelliklerin bir bölümü JAVA Lang. Spec. 3 ile JAVA'ya eklenmiştir.

Java, hedef kod olarak JVM'in (Java Sanal Makinesi - Java Virtual Machine) derleyebileceği byte kodları üretmektedir. Oluşturulan byte kodlar JVM tarafından farklı platformlar için (Intel, RISC vb.) makine diline çevrilmektedir. Dolayısıyla kod optimizasyonu Java derleyicisi tarafından derleme anında yapılmakta, farklı programlama dilleri desteklenmemektedir.

Son yıllarda Dijital Mars şirketinin "d" isimli bir dilden de söz edilmektedir. Bu dil de C# ve JAVA'ya oldukça benzemekle birlikte doğal kod üretmektedir.

C# tür bakımından da daha güvenli bir dildir. Göstericiler olmadığından kullanımı kolaydır.

Diğer .NET Dilleri

C#, .NET ortamının en önemli dili olsa da diğer programlama dilleri de CLI uyumlu versiyonlarını oluşturmaya başladılar.

- <http://www.dotnetlanguages.net/DNL/Resources.aspx>

web adresinden alınan 75 CLI uyumlu dil :

- [A#](#)
- [APL](#)
- [ASP.NET: ASM to IL](#)
- [AsmL](#)
- *Basic*
[QuickBasic for .NET](#)
[VB .NET \(Microsoft\)](#)
[VB .NET \(Mono\)](#)
- [BETA](#)
- [Boo](#)
- [BlueDragon](#)
- *C*
[lcc](#)
[csc](#)
- *C#*
[C# \(Microsoft\)](#)
[C# \(Mono\)](#)
- [Cw](#)
- [C++ \(Microsoft\)](#)
- [Cat](#)
- [CIL](#)
- *Cobol*
[NetCOBOL \(Fujitsu\)](#)

Net Express (Micro Focus)

- CULE.NET

- Eiffel

- F#

- Forth

- *Fortran*

Fortran (Lahey)

Fortran (Salford)

- Haskell (VHS)

- IronPython

- *Java*

J# (Microsoft)

Java (IKVM .NET)

JNBridge

- *JavaScript*

JScript .NET

JANET

- Lego.NET

- *Lisp*

L#

FOIL

RDNZL

- leXico

- LOGO

- Lua.NET

- M#

- Mercury

- MixNet

- Mondrain

- Nermerle

- [Oberon](#)
- [Ook# .NET](#)
- [Pan#](#)
- *Pascal*
 - [Chrome](#)
 - [Component Pascal](#)
- [Pascal \(TMT\)](#)
- *Perl*
 - [PerlNET](#)
 - [PerlSharp](#)
- [PHP](#)
- [Prolog](#)
- [RPG](#)
- *Ruby*
 - [IronRuby](#)
 - [RubyCLR](#)
 - [Ruby/.NET Bridge](#)
 - [Ruby.NET](#)
- *Smalltalk*
 - [S#](#)
 - [#Smalltalk](#)
 - [Smalltalk and .NET](#)
- [Scala](#)
- [SML .NET](#)
- [Spec#](#)
- [Tachy](#)
- [TickleSharp](#)
- [Vulcan.NET](#)
- [Zonnon](#)

Bilinen projeler başlığı altında bir .NET dil listesi de

- <http://www.gotdotnet.com/team/lang/>

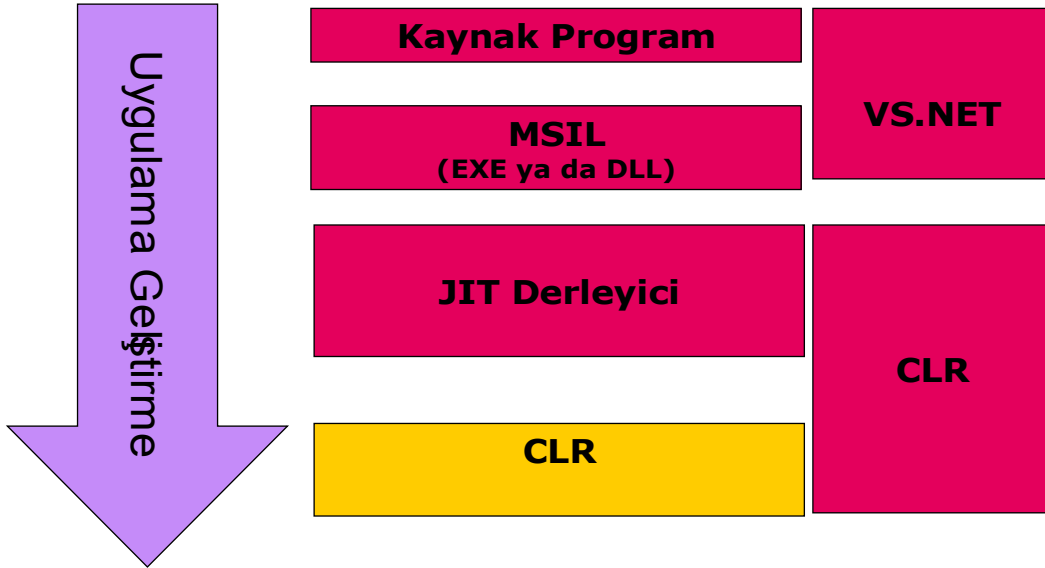
web adresinde yer almaktadır.

.NET KAVRAMLARI

Common Language Infrastructure (CLI) :

European Computer Manufacturer's Association tarafından belirlenen programlama dillerinin ortak standardı. Bu standart aşağıdaki alanları içermektedir :

1. **Common Type System (CTS)** : Tiplerin ve nesnelerin farklı programlama dilleri tarafından paylaşımını sağlayan standart.
2. **Common Language Specification (CLS)** : Derleyicilerin ürettikleri kütüphanelerin yapısını belirten standart.
3. **Metadata**: CLI uyumlu derleyiciler için assembly'ler içinde bulunan modüllerin kendilerini anlatmaları zorunludur. Modüller içerisinde bulunan metadata kullanılan tipleri vb. özelliklerini içeren bilgidir.
4. **Common Intermediate Language (CIL)** : .NET derleyicileri tarafından üretilen, yüksek seviyeli, donanım bağımsız ara dil.
5. **MSIL** : Microsoft Intermediate Language.
6. **IL** : Intermediate Language
7. **Virtual Execution System (VES)** : CIL'in derlemesini sağlayan sistem.
8. **Common Language Runtime (CLR)** : CLI dillerini çalıştıran derleyici.
9. **.NET Framework Class Library** : Geniş ve zengin sınıflara sahip nesne tabanlı kütüphane.



Dökümanı Hazırlayan :

M. Alper SOYASLAN
Elektronik ve Haberleşme Müh.
alper.soyaslan@akademinet.net

Dökümanın Hazırlanmasında Yararlanılan Kaynaklar :

- Kaan ASLAN 'ın C# ders notları
- <http://www.dotnetturk.com/DotNetBasicConcepts.aspx>
- <http://msdn2.microsoft.com/en-us/default.aspx> (Microsoft Software Developer Network)
- <http://www.gotdotnet.com/team/lang/>
- <http://www.dotnetlanguages.net/DNL/Resources.aspx>
- Faruk Çubukçu'nun MCSD .NET (c) Sertifika Programı sunumları
- Yeliz Korkmaz – Mehmet Nuri Çankaya Windows® ve Web Uygulamaları Geliştirme sunumları
- http://www.mono-project.com/Main_Page