

Microsoft's .net Initiative

Hari Sivaramakrishnan

hari@mail.umd.edu

Microsoft .net

How did it all start?

- Development work started in 1998
- First announced to the world in 2000 at the Microsoft PDC
- 2002 saw the first release of v1 of dotnet framework and Visual Studio (Everett).
- 2003 saw the second release of Visual Studio and ASP.net.
- Next release is Visual Studio 2005 (Whidbey).

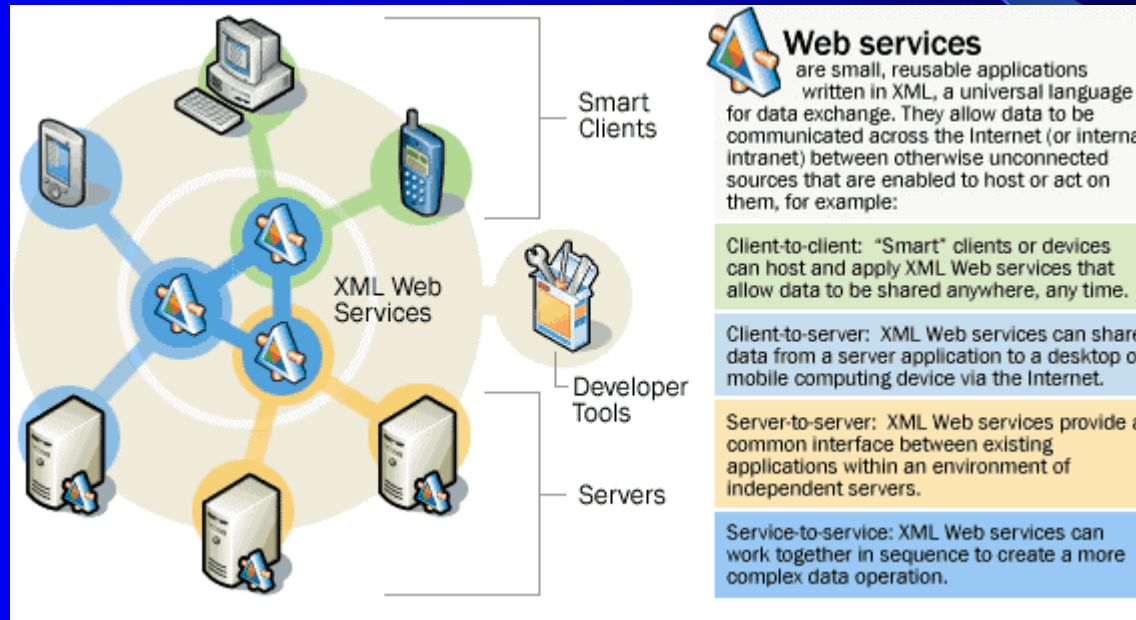
Why .net?

- Microsoft's COM and COM+ were very difficult to understand and convoluted.
- DCOM used to communicate between different systems.
- ASP was a scripting language, this was not good for performance.
- Web Services and XML started to become popular. There was no development platform that had XML tightly integrated into it.

What does .net include?

- Microsoft's company wide initiative.
- The .net Framework includes:
 - Common Language Runtime.
 - Extensive collection of base classes.
 - This is what you need to be able to execute application written using the .net framework on your system.
- Visual Studio.
- Windows Server System, collection of servers to build enterprise applications.

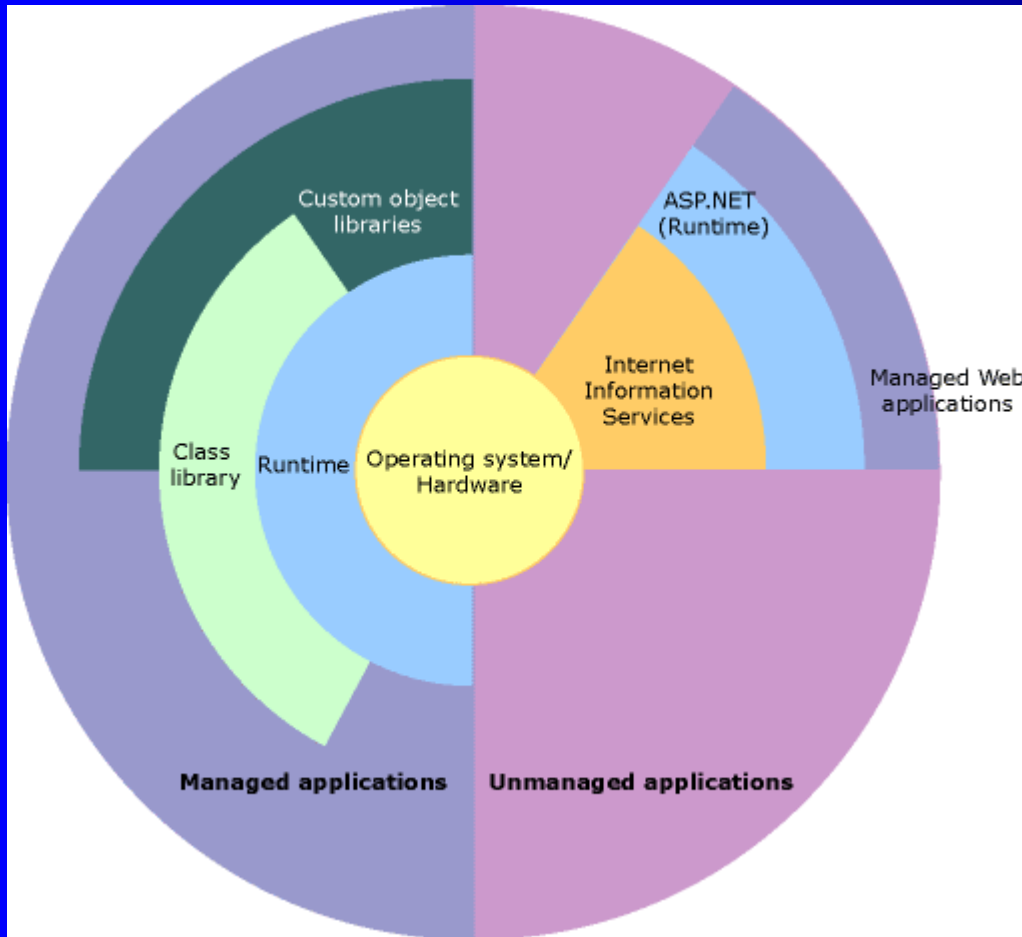
What does .net include?



Some key features of .net framework

- Extensive XML support. A natural addition is support for XML Web Services.
- Any language can be used to write programs as long as they can be compiled down to IL.
- Garbage collector
- JITter
- Managed Runtime.

.net Features



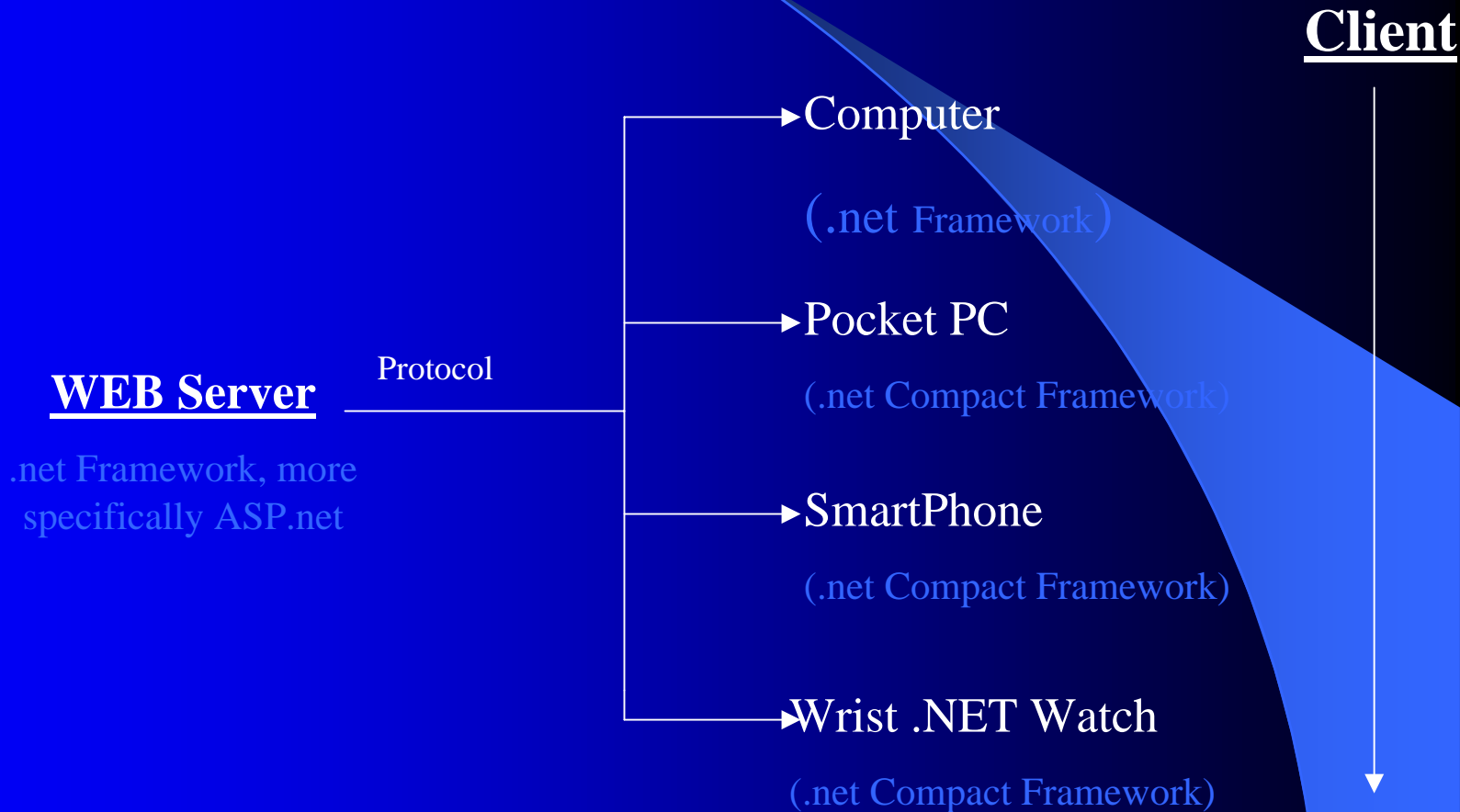
What can you do with .net today?

- Develop web applications or web services with ASP.net, “Web Forms”.
- Develop applications for Windows, “Windows Forms”.
- Develop Simple Console Applications.

Enterprise Application System

- A web application that provides services (XML web services, dynamic websites or just a static website).
- A client application that consumes the services provided by the web application. (A web browser or a more complex app).
- A protocol that enables the exchange of data between the two.

Enterprise Application System



ASP.Net

(<http://www.asp.net>)

- Object oriented way to develop websites.
- Next generation of classic ASP, where websites were created using scripting languages (javascript or VBScript).
- Can use any of the .net implemented programming languages to create websites (C#, VB.net, Managed C++, Perl, Python, Cobol, JScript.Net, J#...)
- An extensive library of “controls” provided by Microsoft. There are several more developed by ISVs.

ASP.Net

(<http://www.asp.net>)

- Clear separation between code and design. This is one of the biggest benefits to using asp.net over classic asp.
- “Code behind” files define the logic and all the operations that are performed on the website; aspx files contain the layout of all the controls.
- All code is executed on the web server, unless specifically made to run on the client by writing the code in javascript.
- Significantly faster than classic ASP, because compiled files are a lot quicker than script files that need to be interpreted every time.

ASP.Net (<http://www.asp.net>)

How do web applications work?

- Web page request is transferred by IIS to the asp.net worker process.
- Code behind file is compiled into .net IL
- IL is “jitted” into native code.
- A DLL is produced. This DLL is used for all future requests to the page unless it has changed.
- The DLL is used to produce an HTML file that is sent to the web browser (For a website, it is different for an XML Web Service).
- All the state information is stored as objects by the worker process, or they can be stored in a SQL server.

ASP.Net

- A single page is made up of “user controls” or “web controls”
- These are just objects that are nested inside the main object, which the “Page object”.
- All the code associated with these “web controls” runs on the server.
- User controls are web controls not developed by Microsoft (instead created by other developers).

ASP.Net

- An aspx page can also have “HTML Controls”, which are the controls that are used in web pages today.
- This is useful when existing ASP code is to be migrated to ASP.net
- In server controls, you are not worried about browser compatibility because all the code is generated on the server. This is an issue with html controls that use client side scripting.

ASP.Net (<http://www.asp.net>)

How do you create a web application?

- Best tool is **Visual Studio (VS)**. (<http://msdn.microsoft.com/vstudio>).
- ASP.net Web Matrix is a free tool provided by Microsoft. This does not allow for separate code behind files like VS does, but takes a more scripting like approach.
- Other tools such as C#Builder (Borland) and Sharp Develop (Open Source) are available. They certainly do not match the standards set by VS.Net.
- Code Demo – How to write an ASP.Net application using VS?
Create a simple web page using C#.

XML Web Services

- The next generation of the internet.
- Uses XML to exchange data over the SOAP protocol.
- Still use a traditional internet connection to send and receive data.
- Code in “web methods” is executed on the web server. Looks just like a traditional method call in the client.
- Some common web services – Passport, MapPoint, Amazon and Google web service APIs.

XML Web Services

- Standards being developed by Microsoft and IBM.
- Locate web services with UDDI.
- Web Services are described using WSDL (Web Services Description Language).

.net and XML Web services

- .net platform was the first that was designed with extensive XML and web services support.
- Windows Server System
- Mobile devices
- Windows codename Longhorn will have web services baked into the OS.
- This is the future of distributed computing.

.net and Web services

How do you consume web services?

- Develop a client application that can be distributed.
- Microsoft Office 2003 makes extensive use of the Amazon Web Services API.
- .net provides extensive support when developing “WinForms Applications” to consume web services.
- Code Demo:
 - Client developed that consumes Google’s Web Services.

.net and XML Web services

How to create web services?

- Visual Studio and ASP.Net provide extensive Web Services support.
- Microsoft provides a directory where you can publicize your web services.
- Code Demo:
 - Develop an ASP.Net web services application using Visual Studio.

.net

Tying it all together

- Build a client application (a WinForms Application) using the .net framework, or use an existing application like Internet Explorer.
- Develop an application on the web server using ASP.Net. This may be a website or a web service.
- Request the website using a browser or through the client application.

J2EE vs .Net

- The two are very similar and it is certain that they will co-exist in the future.
- They seem to borrow the good features from each other.
- Microsoft came out first with its Web Services initiative; IBM and MS are working on the Web Services standards.
- http://java.oreilly.com/news/farley_0800.html

J2EE vs .net

Platform Independence

- Java is known to be platform independent.
- Does this mean .net is platform dependent?

.net Community Effort

- Microsoft's Rotor Project
- <http://research.microsoft.com/Collaboration/University/Europe/RFP/Rotor/>
- ECMA Standardization of C# and CLR
- Mono Project
- DotGNU Project

.net Resources

- [Microsoft Developer Network](#)
- [Employee Weblogs](#)
- Lots of books from the Microsoft Press and other publishers like O'Reilly
- Lots of websites on the internet.