"Technical Aspects of Dot Net and J2EE for Development of Web Applications "

Prof. Arati. D.K

Head of Department, Computer Engineering Dept.

Dhole Patil College of Engineering, Wagholi,

Pune, Maharashtra State, India

Abstract

Any software applications come into two flavors: stand alone applications and web based applications. Standalone applications are usually expensive, especially when we consider the true cost of ownership, which includes high cost items like installation, up gradation of PCs, and training. The Internet boom promised end users efficient access to a host of business critical information on any device, creating a widespread trend towards Web-based applications with browser-based user interfaces. Web based applications allow to link people and data in new and more effective ways, and at a lower cost, in ways that were never possible using standalone software. Web applications are easier to use, and look more attractive than standalone software. Web based application development provides an opportunity to save time and money, and improve the way we interact with clients, suppliers and business partners[5].

Usually web applications widely developed using Java and Dot Net. In this paper author has tried to focus on significance of web applications developed using Dot Net Technology [6].

Web Applications The web applications are steadily increasing its reach beyond the desktop to devices ranging from mobile phones to domestic appliances. This rapid expanding accessibility is largely due to the Web's foundation in open protocols and markup languages, which offer the most widely

Implemented global infrastructure for content and web application access. Web applications are becoming more popular because they offer many advantages over traditional software. Traditional software is usually expensive, especially when we consider the True Cost of Ownership, which includes high cost items like installation, upgrades to PCs, and training. Web applications allow to link people and data in new and more effective ways, and at a lower cost. in ways that were never possible using traditional software. Web applications are easy to use, and look more attractive than traditional software. A web application runs entirely within browser[8][9]. Online shopping is the most common example of a web application.

Web applications add interactivity to a website. Unlike normal website pages that never change, web application pages contain code that make them dynamic. In this paper author has tried to focus on technical aspects of DOT NET and J2EE and their significance in web development.

Keywords: Software, web technology, web applications ,DOT NET, J2EE,IDE.

Web Technology

All web based applications that are available will be designed according to the client's requirement. Every web based product will be having its own merits and demerits. Demerits of any web based product can be gauged depending upon technologies used, level of user friendliness of the product etc. Making a product user friendly depends on the design required by client but speaking on to the technology becomes a choice of product developer as there are plenty of technologies available. The selection of technology depends upon the corporate. Selection of web technology can be made by considering technical aspects of technologies which are used to implement .
Following are few different technologies with which web applications can be developed .

- The web applications implemented on the Java platform where JDBC can be used for data base connectivity with J-Builder to develop the Java classes and JSP (Java Server Pages) can be used for presentation. The whole application can be hosted on Apache-Tomcat web server. Macromedia Dream weaver or Microsoft Front Page can be used to develop HTML templates[4].

- The web application can be implemented on Dot Net platform where ASP.Net and VB.Net can be used for presentation along with ADO.NET (Active X Data Object) as the data access technology with C# (C sharp) coding. Dot Net Framework can be used to develop HTML template.

The fact for selection of technology required to develop an application is generally carried out by considering pros and cons of particular technology[2][3].

Technical Aspects of Dot Net and Java.

The decision to use Dot Net or Java will largely rely on the analysis of the system. Server controls and code behind functions are amazing in Dot Net. Choosing Asp.net has many advantages. In Asp.net we can reduce the code. Asp.net IDE(Integrated Development Environment) has important tools for creating server side controls and easy for validation control. Developers don't need to learn so difficult coding as in J2EE(Java). Asp.net supports event-driven programming in which developer can easily understand and debug the coding in less time. Asp.net is not case-sensitive as java.

In Dot Net, shorter development time could be accomplished, but this could not be achieved with Java, to understand this aspect more consider following example:

With EJBs(Enterprise Java Beans), if any changes were made to a class file or EJB in particular, it would require that the entire application be repackaged and redeployed. With some application servers, we are able to skip the step of repacking by working in an exploded environment, but redeployment or restarting the application server was always necessary. At least this is the case with Web Logic[1].

Company chose the platform for any application definitely depend on how big their budget is and when does the application have to be deployed into production. Everything costs more with J2EE like Hardware, software developers, software licenses (for example, Web logic) cost and because it takes longer to develop a J2EE application, it needs a lot more cash to fund the project.

J2EE application servers are better in terms of performance than dot net applications. But for maintaining the J2EE servers need to invest a considerable amount on hardware. Spending the same amount to upgrade the system with IIS(Internet Information Services: Web Server for Dot Net applications) the performance would be even better.

 Microsoft Dot Net is a programmer or developer-friendly solution, where as the learning curve of J2EE fails to prove that J2EE is programmer-friendly.

Security is the main aspect for everything. So Dot Net will be less secure than their open-source counterparts but this could be overcome using IIS 6.0 when it is implemented in lock mode.

Ultimately we can conclude that Microsoft programming tools are a lot like fast food on a road trip. Hope that Microsoft is unable to destroy the World Wide Web via its Dot Net initiative. Dot Net is simply a proprietary client server application that wraps around web standards (xml, html,etc). Dot Net supports multiple languages that is not possible with Java.

Ultimately the gist is that Dot Net is cheaper, and faster to develop web applications than J2EE.

Conclusion

The best feature of any web-based application, is to have our data centralized in one place and being able to access our data from anywhere in the world, as long as we have an internet connection. The most important benefit which seems to be much appreciated when using web based applications over traditional applications is the no-install process. Hosting and updating of web based application for a company will be just a matter of minutes (compared to years and years with traditional software). Since everybody is running the same version of the application, developers don't need to worry about backward compatibility or similar issues.

Microsoft programming tools have become a boon to develop any complex web based application such that resultant Web application will be fast, user friendly and cheap .

Ultimately we can conclude that Microsoft DotNet provides an challenging and easy   gateway to develop web applications.

References

1. Raghu .R. Kodali and Jonathan Wetherbee, Beginning EJB3.0 Application Development , Apress.
2. Herbert Schdilt, J2EE: The Complete Reference, Mc Graw - Hill Osborne Media, first edition .
3. Stephen Walther , ASP.NET Unleashed, Sams, first edition, 2001.
4. Michael A. Kittel and Geoffrey T. Leblond, ASP.NET 2.0 Cookbook, O'Reilly, second edition, 2005.
5. Grant Palmer, C# Programming Reference, Wrox Press, first edition, 2002.
6. Charles Petzold, Programming Windows with C#, Microsoft Press, Har/Cdr edition , 2001.
7. Hethe Henrickson and Scott Hofmann , IIS 6.0: The Complete refrnce , Mc Graw- Hill Osborne media, first edition, 2003.
8. William R. Stanek, Microsoft IIS 6.0 Administrator's Pocket Consultant , Microsoft Press, 2003.
9. Kalani Schenken, Lee Gibbs, Milner Bell and Clark Homer , ASP.NET 1.0 with C# , Vol. 1.0,Wrox,2002.
10. Joe martin, Brett Tomson,   ASP.NET, Sams, first edition, 2002.
11. Paul ballard, Web Service Performance: A Comparison of J2EE and   .NET ,.NET Journal, 2006.
12. Karim Hyatt, N-Tier Application Development with Microsoft .NET, 2006
13. Lara Ashmore , A Web-based Application Design Overview, IT Journal On-Line: Spring ,1997