WEB APPLICATION TECHNOLOGIES

Most modern businesses software is web-enabled. This article describes how such applications are implemented using the five main technology stacks.

Types of Web Applications

There are three main types of web applications:

- **Customer-facing** applications are known as e-commerce or B2C sites and use the **internet**. These typically present a customer with choices of products or services to buy using a shopping cart and payment method. Examples are travel-reservations, http://www.amazon.com and http://www.ebay.com. This market has really taken off. The Economist (March 13, 2004) estimates that the 200M Americans with web access will buy about $120B of goods and services over the internet this year ($600/person).

- **Employee-facing** applications use the **intranet** in a company. One example is a company's accounting application. Another might be employee expense reporting. A third might be the ERP (enterprise requirements planning) system. These applications previously operated on an internal client-server network. They are now web-enabled to make them easier to use and deploy. Disparate applications, such as an ERP and CRM (Customer Relationship Management) systems are now being integrated using XML and web services.

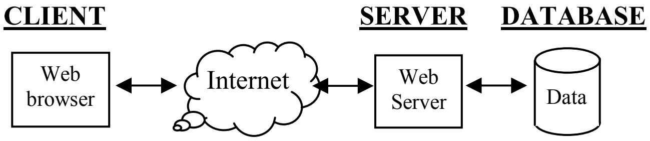
- **Customer-Supplier facing** applications are known as B2B (Business to Business) sites and use the **extranet**, (an extension of an intranet that allows outside companies to work in a password-protected space). B2B sites provide a secure means for sharing selected information. One example is supply chain software that allows all suppliers to see demand and inventory in the supply chain. Another example is procurement software that allows a customer to send RFQs and receive quotes over the web. A third example is collaboration software that allows companies to share product development and project management information.

Not all applications fit the above categories. For example Yahoo! email is not in any of the above. However, the above categories are representative of the main types of applications. The same technologies are used for these other applications.

The 3-tier architecture

Web applications are built using a 3-tier architecture in which the client, server and database constitute the main elements. This is sometimes called an n-tier architecture because there can be multiple levels of servers.

This architecture is distinct from prior mainframe (1-tier) and client-server (2-tier) architectures. The basic structure of a 3-tier architecture is:



Technologies Used to Build Web Applications

Originally, the internet was designed to serve "static" pages. A rudimentary technology based on CGI was developed to allow information to be passed back to a web server. During the last ten years, four main technologies have emerged to replace CGI and the basic CGI technology has been further refined, using Perl as the primary programming language. This has lead to 5 competing technology stacks that differ in the following attributes:

- Programming languages (Lang)
- Operating system (OS). This can be Linux (L), Unix (U) or Windows (W).
- Web server (Server)
- Database support (DB)
- Sponsoring companies (Sponsors)

The following table summarizes these technology stacks. Note variations have been left off to show the main combinations.

Stack	Sponsor	OS	Server	DB	Lang
CGI	Open source	L/U	Apache	Varies	Perl
ColdFusion	Macromedia	W/L/U	ColdFusion	Varies	CFML
LAMP	Open source	L/W/U	Apache	MySQL	PHP
Java/J2EE	Sun, IBM	L/U	J2EE	Varies	Java
.NET	Microsoft	W	ASP.NET	SQL server	VBasic C#

These technologies are quite different, which means that someone who's familiar with one approach would have a high learning curve to use a different one. Once an application is developed using one technology, it is difficult and expensive to convert it to a different one. As a result, many web application developers have a strong interest in promoting the technology they are familiar with.

When choosing a technology for the web application, one typically chooses a whole stack. Trying to use one piece from one stack (e.g. Java/J2EE) and another from a second (e.g .NET) is possible, but difficult--a bit like trying to match plaids and stripes in your wardrobe.

The following is a more detailed explanation of each of these technology stacks:

- **CGI/Perl**. CGI is the granddaddy of interfaces for passing data from a submitted web page to a web server. Perl is an open-source language optimized for writing server-side applications. Together, CGI and Perl make it easy to connect to a variety of databases. Apache tends to be the web server used because it runs on all major operating systems and is highly reliable. Other open-source languages such as C and Python can also be used. For high-end applications, especially e-commerce sites like amazon.com this technology is used because it is so powerful. However other technology stacks can be implemented more easily and quickly.

- **Macromedia** sells a collection of products that make it easy to build small and medium-sized web applications. The primary tools provided by Macromedia are ColdFusion, which is an engine that lets one program in CFML (Cold Fusion Markup Language) and Dreamweaver, which is a development tool for making web applications. Because Macromedia is a smaller player, they have focused on trying to make their products compatible with components from other technology stacks. Macromedia also sells Flash and has tools for using this in web applications.

- **Java/J2EE** is a robust, well-developed method for creating medium to large web applications. It has support from a number of large industry players. Sun Microsystems provides the Java. IBM (Websphere) and BEA Systems (Weblogic) are two major suppliers of web application servers and associated software to make it easy to create and manage these applications. There is a large body of Java programmers available to write the code. This technology stack works with a variety of databases and is particularly well-tuned to mainstream commercial databases like Oracle and DB2. IBM has developed a development environment called Eclipse that is making it easier to write applications, but in general, Java is associated with powerful applications built by capable programmers.

- **LAMP** (**L**inux, **A**pache, **M**ySql, **P**HP) is a relatively new technology stack for building web applications that has been adopted for many small and medium-size web tasks because: (a) the entire technology stack is available through open-source; (b) it works well; (c) it is easy to learn; (d) it allows one to build a web application quickly; and (e) there are many open source code samples that can be bolted together to make a full solution. LAMP relies on CGI for data exchange between the server and browser, but the CGI commands are hidden from the developer. LAMP doesn't have all of the capabilities of J2EE, but it gains ground every year. Sites that use PHP can be seen by the ".php" as part of the page name in the URL. LAMP has become especially popular with ISVs (independent software vendors) because they can create an application and sell it without having to pay for the underlying (open source) software.

- **Microsoft .NET**. Microsoft is using their .NET strategy to take over the server market the way in which Windows, Office, and Internet Explorer have taken over the desktop. The stack comprises a web server (ASP.NET) and two programming languages (VisualBasic.NET and C#.NET) that compete against PHP and Java respectively. They also have a database (SQL Server). Microsoft has done an excellent job making their products easy to use so a business analyst can create a web application without needing a programmer.

Conclusion

I hope that you have found this brief helpful. Please let me know if you have any comments, corrections, questions or suggestions for areas that should be covered in more depth. The following are two good references if you'd like to explore this topic in greater depth:

- Troller, Michael. *How the Internet Works.* (c)2002 Que.

- Vermeulen, Bert. The Corp21.com site as several related articles on topics such as "*Making Your Website Popular*", "*Promoting your website Through Search Engines*" and "*Web Basics*."

- Zhao, Jenson J. *Web Design and Development for eBusiness*. (c)2003 Prentice Hall.

About the Author

Bert Vermeulen owns Corp21, a company that supports, incubates and advises businesses, entrepreneurs, and inventors around the world. For more information, see http://www.corp21.com.