Leveraging Client/Server Architecture Within the Enterprise

Edgardo Donovan
Touro University International
ITM 506
Dr. Irene Tsapara
Dr. Wenli Wang
Module 2 – Case Analysis
Monday, November 6, 2006

Leveraging Client/Server Architecture Within the Enterprise

" *The term client/server was first used in the 1980s in reference to personal computers (PCs) on a network. The actual client/server model started gaining acceptance in the late 1980s. The client/server software architecture is a versatile, message-based and modular infrastructure that is intended to improve usability, flexibility, interoperability, and scalability as compared to centralized, mainframe, time sharing computing.*" (**Sadoski**)

Client/server architecture has emerged as the de-facto standard in worldwide computing networks surpassing the mainframe computer after the technological advances of the mid 1970's which led to standardized operating system and networking protocols now widely accepted throughout the home and business computing industry.

Mainframe computers are generally standalone machines which contain an operating system, a series of files, and application designed to process them. In order to use the mainframe typically a user would interface directly with it via a keyboard and a monitor. Mainframes used to be the de-facto standard for the government, large corporations, and large universities prior to the advent of personal computers along with the operating system/networking standards we have today. File sharing among mainframe computers was prevalent before that time when collaborative computing started becoming feasible financially and technically.

"*As a result of the limitations of file sharing architectures, the client/server architecture emerged. This approach introduced a database server to replace the file server. Using a relational database management system (DBMS), user queries could be answered directly. The client/server architecture reduced network traffic by providing a query response rather than total file transfer.*" (**Sadoski**)

Eventually demand for ubiquitous file sharing within large networks large up to thousands of nodes was realized via client/server network architectures. Client server involves one computer within a network acting as file sharing agent. Usually these computers have different server customized operating systems. In a client/server architecture certain files or objects within a network reside on a server computer and are given an IP or network address. Each time that clients or workstation computers within that network make a request for a file or object residing on the server a request is made and the server responds by sending that file to the client. Within such an architecture additional permissions and business rules can be set to allow different types of access to different users. Some users may have administrative read/write permissions whereas others may only have read access.

" *Incompatible versions of Unix, SCSI, multitudes of peripheral device drivers, what have you cause much frustration. In my opinion things are too complex today because of the total lack or disregard of standards. Let's face it, it was a challenge making it work when it was all blue. Now with the LAN from one vendor, the server from another, and software from a multiple of sources it is no surprise that making it work is so difficult. Welcome to the multi-vendor world of Client / Server.*" (**Janco Associates**)

There are many advantages that make a client/server architecture the leading choice among the majority of corporate networks. Scalability is optimal as a single server may be able to server hundreds of workstations effectively. As networks expand multiple servers can be used together provided that load balancing is ensured. Client/server architecture also makes it easier for network administrators to update and install software on multiple computers remotely. Therefore, maintenance costs of network ownership in client/server systems usually get proportionally smaller as networks grow.

However, there are some cases when it would not be practical to utilize such a system. Scientific teams or classified government research projects may not have the need for extensive file sharing and may prefer the more isolated architecture environment offered by a traditional mainframe or stand alone power workstation environment. They may also not want to deal with the interoperability issues that may arise with the use of more sophisticated client/server operating systems and certain types of high power workstation-centric processing software and related hardware peripherals.

The decision for large established companies to move away from or to upgrade their IT systems in an attempt to better leverage the information currently residing in old legacy mainframe systems is not one taken lightly. Managers need to figure out the necessary realistic returns on IT investments from such changes in order to offset what typically are very large, complex, and ultimately expensive IT overhauls. In order to do so it is necessary to map out a prospective life-cycle plan for current and future IT systems which given the current ever so quickly changing technology environment are hard to assess.

.

"*One more complicating factor is the seeming lack of quality control standards from both hardware and software vendors. In the last several months there are been a number of occasions in which both hardware and software has been released by the industry giants of Client / Server only to have almost universal acknowledgment of major system and component failures. This is compounded by the fact that these same vendors take several months to get "maintenance" releases developed and shipped."* (**Janco Associates**)

There are many different ways in which an organization can decide to change its IT infrastructure in a manner that would give them capabilities over the long-term that their current mainframe legacy systems cannot. One popular option over the years has been the creation of a middle application tier commonly known as "middleware" positioned between old legacy systems and new client/server Windows/Unix operating environments. With the advent of XML it has become easier to take data residing through old legacy systems with the help of middleware applications and render it ready usable through databinding by a variety of browser based interactive database applications.

Client/server architecture has emerged as the de-facto standard in worldwide computing networks surpassing the mainframe computer after the technological advances of the mid 1970's which led to standardized operating system and networking protocols now widely accepted throughout the home and business computing industry.

BIBLIOGRAPHY

I. Works Cited

Sadoski, Darleen. Client/Server Software Architectures. Carnegie-Mellon University, 1997

Janco Associates. Myths of Client-Server Architecture. Janco Associates, 20057

II. Works Consulted

Sadoski, Darleen. Client/Server Software Architectures. Carnegie-Mellon University, 1997

Janco Associates. Myths of Client-Server Architecture. Janco Associates, 20057

Wikipedia. Interaction Design. Wikipedia.com, 2006

Donovan, Edgardo. Online Seminar on Full Life-Cycle Web Presence Management. EddieDonovan.com, 1999

Nielsen, Jacob. Designing Web Usability. Peachpit Press, 1999

NETworldwide.com. Former Antivirus Software Industry. NETworldwide.com, 2005

Download.com. Cheap or Free Antivirus. Download.com, 2005

Vamosi, Robert. Antivirus software must be free. Here's why. ZDnet.com, 2004.

Gaudin, Sharon. IT and End Users Differ on Spam Severity. EnterpriseITplanet.com, 2004.

Gallant, John. Convergence: Hype No More. Network World, 2004

Clark, Drew. Three Forms of Convergence. The Watergate, 2004

Goleniewski, Lili. Understanding the Telecommunications Revolution. SaveDiggDel.icio.us, 2002

Ansoff, Igor. Corporate Strategy. McGraw Hill, 1963

Alfred, Alfred. My Years with General Motors. Currency Doubleday, 1963.

Jackson, Tim. Inside Intel. 1997.

Gates, Bill Business at the Speed of Thought. Warner Books, 1999.

Grove, Andy Only the Paranoid Survive. Currency, 1996.