

Client/Server Computing: The Opportunities and Challenges

Leonard M. Greski
Hewitt Associates

Abstract

In a discipline where change is occurring at an ever increasing rate, it is a challenge for practitioners to keep up with the latest advances in technology. To many SAS® users, the need to access data that resides on a variety of machine platforms is a familiar one. However, those who have tried to deliver client/server applications to coworkers and clients have discovered shortcomings in the current technology, and faced daunting challenges during implementation of the technology.

This paper will present a definition of client/server, an overview of the business need for distributed data access, the opportunities for our organizations that can be gained through successful implementation of the technology, and the challenges we must overcome if we are to successfully build client/server applications.

Introduction

Client/Server computing has replaced Computer Aided Software Engineering (CASE) as the darling of the Information Systems media. Hardly a week passes without at least one article about client/server technology occupying the pages of the major trade

publications.

Why is so much attention being given to client/server technology? Is it due to the sheer number of software vendors that are manufacturing client/server products? Is it because client/server is the long awaited silver bullet that will slay the software engineering werewolf?

No, client/server technology is receiving attention because the complexities of our business activities demand it. Client/server is the next step in the evolutionary process of delivering useful information at the correct level of detail to the right people, so they can make strategic business decisions.

During the past twenty years, we have seen phenomenal increases in the computational power of computers. This increase was accompanied by an equally dramatic decrease in the cost of these machines. Changes in the economics of the computer industry have led to an explosion of application development on an increasingly diverse universe of hardware. In addition, improvements in the design of user interfaces has radically changed the way people interact with the machines, making it possible for people who are not programmers to be productive with them.

While these advances are impressive, the Information Systems community has been only marginally successful at making information the strategic tool that businesses require in order to gain a competitive advantage.

Information Systems staffs are stung by complaints about the dreaded "applications backlog". Application development projects are almost never completed on time, within budget, and to the desired level of performance. In addition, once a system is completed, it goes through a seemingly endless cycle of bug fixes and "enhancements". The bug fix / maintenance cycle consumes up to 70% of the resources expended on a system over its life cycle (Arthur, 1993). Consequently, as the number of delivered applications an organization must support increases, its ability to deliver new applications decreases.

How does client/server technology address these problems? Client/server computing takes advantage of the advances in hardware, software, and user interface design, using them to push information delivery as close as possible to the decision-makers within an organization. It provides a framework distributed storage and processing of information across networks of dissimilar machines.

What Is Client/Server?

Client/server technology can be viewed from two perspectives: the technological perspective and the business perspective. From the technological view, client/server can be seen as the synthesis of seemingly diametrically opposed computing

environments, the mainframe and the personal computer. Client/server architecture calls for some level of cooperation between a local machine (the client), and a remote machine (the server). These machines communicate with each other over a communications network.

There are three types of functionality that must be accounted for when building a client/server application: the user interface, the application logic, and data access (Wallace, 1992). There is a significant amount of variation in the ways these functions can be organized in a client/server application. In fact, the Gartner Group has described five separate models of client/server computing, ranging from a server-centric model where only the user interface is controlled by the client machine, to a client-centric model where all functions except data access are controlled by the client (Winsberg, 1993).

From the business perspective, client/server technology is the way that information technologists will provide access to strategic data to the right people in the organization in a timely fashion. In order to achieve this business objective, client/server technology must provide for ease of use, cost-effectiveness, application flexibility.

Ease of use means that business people must have access to information that is displayed at a level of analysis that is appropriate to the business issues being addressed by the information. It must be neither too highly summarized nor displayed in too much detail. In addition, business people need to

interact with data using techniques and methods with which they are already familiar, such as the spreadsheet metaphor. Therefore, client/server technology must provide user interface tools that are robust enough so that application programmers can develop user interfaces that allow business people interact with their data in ways that facilitate strategic decision-making (Rash, Jr., 1993).

Cost effectiveness requires that a client/server solution be less expensive than both a mainframe-only solution and a PC-only solution, not only in terms of application performance, but also in terms of development and maintenance cost. For client/server to become a dominant technology, it must provide for the rapid assimilation of client/server skills by mainframe and PC developers alike. Development cost savings must be shown over the entire software development life cycle (Ross, 1993).

Finally, flexibility must be achieved by delivering information to the business people regardless of the type of client machines they use, as well as being able to deliver the information to multiple locations throughout the geographic areas where the company does business.

The Three Tradeoffs

In addition to the business and technological definitions of client/server technology, there is one imperative given to application developers, regardless of the implementation model being considered: *minimize use of precious*

computing resources.

In a client/server implementation, there are three resources whose consumption must be controlled: processing, data access, and data transfer. Minimizing the overall usage of these three resources is a very complex task, requiring detailed knowledge of the costs of processing and data access on both the client and the server, as well as the cost of sending data back and forth from one machine to the other.

Consider the mainframe-centric client/server model. Under this scenario, the client machine is almost reduced to being a dumb terminal, as its only processing responsibility is to manage the user interface. This model of computing increases hardware costs (a workstation is more expensive than a terminal) without significantly reducing processing on the server.

Conversely, under the client-centric scenario, the server is only responsible for data access. Therefore, all data requested by the client must be moved to the client in order to be processed. Given the huge volumes of data that reside on mainframe-based servers, it is likely under this processing model that client machines would quickly become overwhelmed by an onslaught of data. In addition, it is easy to imagine a scenario where the client-centric model would result in an unacceptable level of network traffic, as huge volumes of data would be sent to the client only to be dropped from the result set of a query.

Clearly, efficient use of computing resources requires that the correct resources are used when appropriate.

This generally means that applications must not only know *how* to process the information, but also *where* to process it. Hardware and software vendors are working at a frenzied pace to build tools that facilitate the intelligent distribution of data access, application logic, and user interface among multiple machines, but to date no single vendor has provided a complete solution to the problem.

Conclusion

Businesses in the 1990's require timely information to gain competitive advantage in a global economy. Client/server technology is seen as the vehicle that will provide the right information to key decision-makers at an appropriate level of detail in a timely manner. The opportunity is here for information technologists to harness the advances in hardware, software, and user interface design and finally deliver on the promise that information is of strategic value to a company.

However, we face daunting challenges in making the client/server dream a reality. They include coping with increased complexities in the hardware, software, and communications networks required to support distributed data access.

In human terms, client/server technology requires us to train thousands of application developers how to minimize consumption of processing, data access, and communications resources while developing an application that has a pleasing user interface and is of strategic value to the corporation.

Can vendors deliver client/server development tools that will enable developers to write portable applications quickly, reduce maintenance, minimize resource consumption, and deliver strategic information to the right people? Only the vendors' R&D staffs and the choices we make with our information technology budget dollars will determine the answer to this question.

References

- Arthur, Lowell J., *Improving Software Quality: An Insider's Guide to TQM*, Wiley & Sons, 1993.
- Rash, Jr., W., "Six Ways to Begin One Company's Migration", *Corporate Computing*, June 1993.
- Ross, N., "Client/Server Methodologies: A Roadmap for Success", *CASE Trends*, March 1993.
- Wallace, J., "Maximizing Computer Resources with Cooperative Processing and Distributed Data Access", *Proceedings of the Seventeenth Annual SAS Users Group International Conference*, Cary NC: SAS Institute Inc., 1993.
- Winsberg, P., "Designing for Client/Server" *Database Programming and Design*, July 1993.

Acknowledgements

SAS® is a registered trademark of the SAS Institute, Inc. in the USA and other countries. ® indicates USA registration.