Client Server Technology

Prepared by Richard W. Boss

Client/server technology is the computer architecture used in almost all automated library systems now being offered to libraries. The simple definition is:

> Client/server is a computer architecture that divides functions into client (requestor) and server (provider) subsystems, with standard communication methods (such as TCP/IP and z39.50) to facilitate the sharing of information between them.

Among the characteristics of a client/server architecture are the following:

- The client and server can be distinguished from one another by the differences in tasks they perform
- The client and server usually operate on different computer platforms
- Either the client or server may be upgraded without affecting the other. Clients may connect to one or more servers; servers may connect to multiple clients concurrently
- Clients always initiate the dialogue by requesting a service.

Client/server is most easily differentiated from hierarchical processing, which uses a host and slave, by the way a PC functions within a system. In client/server the PC-based client communicates with the server as a computer; in hierarchical processing the PC emulates a "dumb" terminal to communicate with the host. In client/server the client controls part of the activity, but in hierarchical processing the host controls all activity. A client PC almost always does the following in a client/server environment: screen handling, menu or command interpretation, data entry, help processing, and error recovery.

The dividing line between the client and a server can be anywhere along a broad continuum: at one end only the user interface has been moved onto the client; at the other, almost all applications have been moved onto the client and the database may be distributed. There are at least five points along the continuum:

Distributed presentation:
The presentation is handled partly by the server and partly by the client.
Remote presentation:
The presentation is controlled and handled entirely by the client.
Distributed logic:
The application logic is handled partly by the server and partly by the client.
Remote data management:
Database management is controlled and handled entirely by the server.
Distributed database:
Database management is handled partly by the server and partly by the client.

An example of one client communicating with several different servers is a PC-based client which can access a library's own automated library system, a CD-ROM server, the automated library system of another library, or a remote online reference service.

While a client usually is configured on a PC and a server on a supermicro, mini, or a mainframe computer; a single machine can act as both client and server on a network. For example, two automated library systems in different libraries that are linked for resource sharing are functioning as clients when requesting information, and as servers when providing it.

There are, therefore, two major applications for client/server in a library environment: 1) as the architecture for an automated library system, and 2) as an approach to linking heterogeneous systems. In the first application, a vendor designs a system using client/server architecture to facilitate use of that system to access multiple servers, to facilitate bringing together multiple product lines, and/or to improve productivity. In the second application, a vendor designs a client to facilitate transparent access to systems of other vendors, and a server to facilitate transparent access to its system from others. While the underlying principles are the same, the vendor has considerable latitude in the design of its own client/server system, but must strictly conform to standards when using client/server to link its system with those of other libraries.

While it has been possible to access a wide variety of electronic resources through an automated library system for a number of years, client/server technology has made it possible to tailor the user interface to provide a personalized interface which meets the needs of any particular user based on an analysis of tasks performed or on an individual's expressed preferences. An example of this tailoring is the recent introduction of portals, common user interfaces to a wide variety of electronic resources with the portal. [See the Tech Note on Portal Technology]. The portal can be tailored to groups of staff or patrons, or to each individual.

Vendors with multiple product lines can build a single client to work with any of their server products. This substantially reduces development costs.

Client/server can also improve productivity. Many vendors are now offering different clients for technical services, circulation, and patron access catalog applications.

A GUI (graphical user interface) -- a presentation of information to the user using icons and other graphics -- is sometimes called client/server, but unless information moves from the server to the client in machine-readable (raw) form, and the client does the formatting to make it human-readable, it is not true client/server. Further, there is nothing in the client/server architecture that requires a GUI. Nevertheless, most vendors of automated library systems use GUI for staff applications. The GUIs are proprietary to each vendor. Web browsers are preferred for patron applications because they are more likely to be familiar to them than a proprietary GUI.

An important computer industry development which has facilitated client/server architecture is referred to as "open systems" C a concept which features standardized connectivity so that components from several vendors may be combined. The trend to open systems began in the 1970s as a reaction against proprietary systems that required that all hardware and system software come from a single source, and gained momentum in the 1980s, as networking became common. While various parts of an organization might not hesitate to purchase proprietary systems to meet their own needs, the desire to provide access from other parts of the organization, or to exchange information, would be an incentive to select an open system. For client/server, open systems are essential.

Most client/server systems offered by automated library system vendors use an open operating system such as UNIX or one of its variations, or Windows NT or 2000 server. UNIX is the most popular operating system for servers because of the large range of platform sizes available, but Windows NT or 2000 server has been growing in popularity, especially for systems supporting fewer than 100 concurrent users.

The most popular client operating systems are Windows 95/98/Me/2000 and Linux. By supporting multiple client operating systems, a vendor of an automated library system makes it possible for the client to conform to a staff member's or patron's accustomed operating system environment.

Almost all client/server systems use a relational database management system (RDBMS) for handling the storage and retrieval of records in the database using a series of tables of values.

There is a common misconception that client/server is synonymous with networked SQL (Structured Query Language) databases. SQL, a popular industry-standard data definition and access language for relational databases, is only one approach -- albeit the one selected by almost all automated library system vendors. While one can reasonably expect the use of an RDBMS and SQL, the absence of either does not mean that a system is not client server.

Client/server requires "intelligence" (e.g., memory) on the client, therefore, most clients are PCs. A "dumb" terminal won't work because it lacks memory, but a "thin client" device will because it has memory and can download applications from the server and do local processing. Although the term "thin client" originally referred to software, it is now increasingly used for computers, such as network computers and PDAs (personal digital assistants).

A network computer is a PC without a hard disk drive. They have been little used in libraries because most libraries use their PCs for a variety of applications in addition to accessing the automated library system. Even when there are applications that lend themselves for use on thin clients, most libraries have preferred to use older PCs that are no longer suitable for applications that require robust machines. They use a two-tier PC strategy that involves the purchase and deployment of new PCs for applications that require robust machines and redeployment of the replaced machines for applications that they can support. For example, new PCs are used for most staff applications and patron access to the Internet: older PCs are used as "express catalogs," devices that have a Web browser, but are limited to accessing the library's patron access catalog. The two-tier PC strategy can extend the life of a PC by as much as three years.

Network computers are most widely used in large organizations that have to support thousands of users. Almost all applications, including word processing, spreadsheets, and other office applications, are loaded on a server. It is then not necessary to load new product releases on each machine; only the applications on the server have to be updated. Most libraries do not have enough users to realize significant savings by taking this approach. For libraries that have hundreds of PCs, remote PC management is an alternative to thin clients; for libraries that have fewer than 100 PCs, it is possible to support each individually.

A PDA is a handheld device that combines computing, telephone, fax, and networking features. While originally pen-based (i.e., using a stylus), many models now come with a small keyboard. The Palm Pilot is an example of a PDA. A number of libraries now encourage their users to access the patron access catalog with a PDA. It is this application which holds the most promise for the use of thin clients in libraries. As the bandwidth available for wireless applications increases and the costs of PDAs drops, the use of PDAs for access to databases is expected to increase dramatically.

The key to thin client technology is Java, a general purpose programming language with a number of features that make it well suited for use on the Web. Small Java applications are called Java applets and can be downloaded from a Web server and run on a device which includes a Java-compatible Web browser such as Netscape Navigator or Microsoft Internet Explorer. This means that the thin client does not need to be loaded with applications software.

A thin client can also be GUI-based. In that case, the client handles only data presentation and deals with the user interaction; all applications and database management is found on the server. Vendors of automated library systems favor proprietary GUI-based clients for staff because that makes it possible to exploit the features of their systems.

February 2002