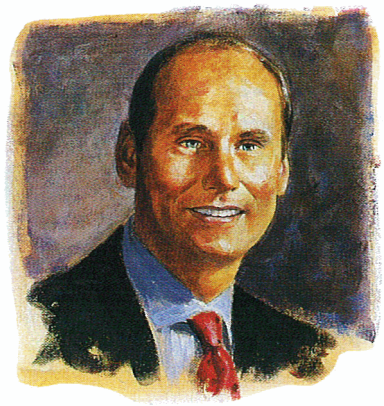
PERSPECTIVE

Downsizing With Client-Server Computing

Dr. George Schussel is the president and principal consultant of Digital Consulting Inc. (DCI), Andover, MA. He is the chairman of the DCI Management Conference on Downsizing, to be held August 13-14, 1990, in Boston, Massachusetts.



JIM LA MARCHE

Downsizing—moving from a few large computers to many smaller platforms—is one of today's most important business trends. It has many benefits, the most obvious being an enormous reduction in hardware costs. The cost for one million instructions per second (one mips) on small platforms is $500-$2,000 per mips, compared with more than $100,000 per mips on large mainframes. If downsizing enables an organization to replace a $2 million mainframe with half-a-dozen $10,000 PCs or workstations, the savings are extraordinary.

Not surprisingly, lower hardware costs are accompanied by lower software prices. The software for a modern distributed network of workstations is significantly cheaper than software of comparable capability on minis or mainframes. Today, organizations can purchase microcomputer and LAN operating systems, a graphical interface, and a window-based 4GL with a companion relational DBMS—all without sacrificing substantial computing power.

Moving Into Open Systems

When organizations move their information management needs to PC and workstation platforms, they're into "open systems" territory. The large number of suppliers and new technologies brings many vendors to these smaller platforms. Price and service competition is plentiful, and the user who commits to downsized platforms will enjoy significant vendor independence.

The resulting savings in people and administration costs are harder to document than are hardware and software costs. Certainly, the technical challenge of designing a good SQL database is not going to vary significantly depending on whether it runs on a PC or a mainframe. When companies downsize their computer systems, however, they can sometimes save money on staffing requirements; employees with PC expertise might demand lower salaries than would those with mainframe expertise.

Client-Server Computing

One of the best ways to downsize is by using the new generation of SQL-based client-server computing technologies from vendors such as Oracle, Sybase, Gupta and Novell. In the client-server model, the application is split between functions that execute on the client, a PC or workstation, and functions that run on the server, a multiuser data repository. Most application logic runs at the client desktop machine. When the application requires data, it generates the necessary SQL command and then passes high-level code to the communications facility. This facility then directs the SQL commands to the server, where the database request is executed.

The idea of managing data on a separate machine fits well with the management approach of treating data as a corporate resource. In addition to executing the SQL statement, the server handles security and provides for concurrent access to the data by many queries.

A benefit of using SQL client-server computing is that the hardware and software products supporting this approach are new and take advantage of the latest developments, such as application languages in a windowing environment. Another benefit is network efficiency. In traditional file-serving PC LAN approaches, the entire data file must be transmitted across a network to the client machine. With SQL as the basis for database management, this problem is resolved, since only the necessary query response data (a table) is transmitted to the client machine. SQL on the server also enables the implementation of advanced facilities, such as triggers and automatic procedures in the database.

The largest market for client-server computing is likely to develop with a mix of OS/2 and PC-DOS as the client and either UNIX or OS/2 as the base for the server. OS/2- or UNIX-based SQL server software will provide security, recovery, and data integrity at mainframe-level capability. Functions such as automatic locking and commit rollback logic, along with deadlock detection and a full suite of data administration utilities, are expected on the server

side. Another advantage of this architecture is that SQL client-server technology enables PCs to be made into "industrial-strength" computing engines.

Performance

Users who haven't built PC-based database applications in the last few years may be suspicious of claims that PCs are capable of performance comparable with that of minicomputer technology. However, the processing capability of a typical PC has increased tenfold from 1984 to 1990. A 33Mhz, Intel 80386-based PC has 30 times the computing power found in a PC XT. Benchmarks audited by Digital Consulting have shown that a 80386-based PC can handle about 10 debit-credit transactions per second while running under OS/2. This level of service can provide on-line transaction processing capability at a cost of $5,000 per transaction/second (TPS)—which is much less per TPS than existing minicomputer and mainframe systems can provide.

With proprietary minicomputers, users can expect to spend from $25,000 to $40,000 per TPS. IMS-based MVS mainframe environments typically yield a cost of $50,000-$75,000 per TPS. Alternatively, using the combination of MVS and DB2 as a transaction processing engine will typically result in a cost of $125,000 per TPS. Based upon full development, maintenance, hardware, software, and staff costs, SQL client-server computing is likely to result in finished systems that cost only a small fraction of what transaction systems have cost in the past.

Vendors have begun to combine microprocessor-based technologies with high speed buses, channels, and parallel computing architectures to create platforms that can run with the fastest minicomputers. Companies such as Compaq, Pyramid, and Sequent are building parallel processing machines using complex instruction set computing (CISC) or reduced instruction set computing (RISC) microprocessor units capable of reaching a sustained processing capability of dozens of mips. In the future, these new hardware systems—together with software from companies such as Microsoft and Oracle—will deliver computing technologies comparable to IBM's largest machines, at a fraction of the price.

Advantages of Client-Server Computing

Besides reducing costs on the hardware/software platforms, client-server computing offers many other important user benefits:

- Developers can use PCs instead of timeshare terminals as a primary development platform
- Even though the PC is used as a principal platform, users can expect security, integrity, and recovery capability comparable to that provided by minicomputers
- SQL's efficient query and transmission facilities greatly reduce the network communication load
- Gateway technologies, an important component of client-server computing software, will allow PC users to gain access to data located in mainframe and minicomputer DBMS products such as DB2, IMS, and Rdb
- The client-server model isolates the data from the applications program in the design stage, allowing flexibility in managing and expanding the database and adding new programs at the application level
- The client-server model is very scalable: as users require more processing power, they can add more servers to the network or "trade up" their servers to the latest generation of microprocessor
- SQL offers users a great deal of flexibility, since it has been adopted by almost every vendor as a relational DBMS standard. Commitment to a SQL server engine will mean that most front-end 4GL, spread-sheet, word processing and graphics tools will be interfaced to the SQL engine

Caveat Emptor

The combination of client-server and downsizing sounds like a technology almost too good to be true. In fact, the wary buyer should know of many pitfalls on this road.

First, client-server *is* new technology. Even major vendors have stubbed their toes bringing it to market. As users decide to acquire software, they should check references carefully and initiate new projects step by step.

Second, many companies are taking an approach to client-server computing based solely on PC LANs. LANs may not offer the same level of uptime and reliability as do terminals hardwired to a minicomputer. Recent statistics seem to indicate that PC LANs will have two outages per month, each one lasting an average of five hours. Other surveys indicate that 96 percent uptime is not unusual for PC LANs—compared with minicomputer environments that are up 99 percent of the time.

Third, since software products to support client-server computing are new, users can purchase only a few finished applications; consequently, they will probably have to write their own applications.

Finally, selling and installing client-server software is much more complex than dealing with traditional PC software. Selling and supporting this new technology has overwhelmed the traditional retail channel. At the same time, traditional large vendors such as IBM, Honeywell, and Unisys haven't yet developed a serious commitment to supporting client-server computing. As a result, users should plan on building their own staff expertise. Users who base their implementation on LAN technology should expect to employ one full-time systems support administrator for every 10-15 PCs on the LAN. In addition, their database administrator requirements will be comparable to those in minicomputer environments.

What to Do Next?

One of client-server computing's biggest strengths is that it is an easy technology to migrate toward—users don't have to throw away their current investment in computer

systems. Users should look at the additional functionality in client-server computing that can extend and complement their existing systems. As a first application, for example, users could employ a client-server computing model for decision support which would use read-only capability against data located in a mainframe or minicomputer database. Offloading significant computing cycles for the decision support application would result in important savings. Later, users could implement plans for true cooperative processing or transaction-based applications that would have gone on mainframes.

The critical new strategies of the 1990s will involve cooperative processing, downsizing, and parallel processing. All of these technological trends are important components of SQL client-server computing. The benefits of this new approach are many, and should be analyzed for inclusion in the systems plans of modern companies. Any company that ignores downsizing trends risks giving up competitive advantages to others.

Ultimately, SQL client-server computing will bring transaction processing automation advantages to environments that have not previously enjoyed computer support. Companies that commit to this approach are buying into a new technology at an early point in its life cycle—and will reap the rewards for the next decade. ■