Front-end transaction processing in the brokerage industry

by ALFRED S. KROIN

Systems Imagination

Jericho, New York

ABSTRACT

Many brokerage systems, such as price-and-quotation display and decision support systems, depend on the timely receipt of pricing information about stocks and other traded commodities. Often these systems subscribe to data services provided by outside sources, and as such are susceptible to reception delays or vendor system malfunctions. This paper presents the alternative approach of user-owned data lines and computer equipment. The relative merits of several systems are discussed, including various aspects of hardware selection and software design.
INTRODUCTION

As is the case with almost every other industry, the past ten years has seen a marked increase in the use of computer systems by the brokerage community. Although these systems have automated widely diversified applications, many share a common basis from which their utility is derived: the prices of marketed securities. Consider the following types of systems: price-quotation display, portfolio management, arbitrage and options analysis, decision support, and historical and archival maintenance. Be they concerned with stocks, bonds, options, commodities, money markets, a combination of these, or another tradable entity, each system depends on the timely receipt of information about the value of its principal items of interest.

In most cases, depending on the degree of timeliness required, the data to drive these systems come from one of two types of external sources. They can be purchased from a "quote vendor" or other communications-based feed (typically a service bureau) if data are needed in real time. If less timely reporting is acceptable, a magnetic tape or other end-of-day service may be purchased. These systems have several disadvantages that exist in varying degrees, depending on the specific data source, including the following:

1. The data are not timely enough. Real-time data may be near real-time, and end-of-day data often are perceived as yesterday's news. On busy trading days, the delivery of data may run behind the actual market position. This can affect the functioning and possibly even the validity of the system.

2. The data may lack flexibility. Depending on the transmission media and the service, the data may not be provided in a form flexible enough for the specific user's needs. Data displayed on a vendor-connected CRT, for example, cannot be relayed to an in-house computer for other calculations or database maintenance.

3. There may be a lack of reliability. Vendor equipment (both on-site and off-site) is beyond the user's control. Regardless of the degree of reliability and fail-safe mechanisms put in place by the user, he is still susceptible to vendor equipment malfunction.

4. Cost factors may be involved. Most services are purchased for monthly fees on an annual basis. The user has no equity and must pay for the services and facilities indefinitely. Fees also are normally proportional to the number of locations and terminals attached; as the system grows, so does the cost of the data.

The alternative to purchasing the data from a middleman is to subscribe directly to one or more of the data communications services offered by the exchanges themselves. There are several widely used services: The Consolidated Tape System (CTS) gives trade prices of the New York and American Stock Exchanges. It is offered by the Securities Industry Automation Corporation (SIAC). The Consolidated Quotation Service/Best Bid and Offer (CQS/BBO) system provides New York and American stock quotations through SIAC. The Options Price Reporting System lists trade and quotation data on stock options as offered by the Options Price Reporting Authority (OPRA). Finally, the NASDAQ Level 1 and National Market Service (NMS) provide quotations and trading information for stocks carried by the National Association of Securities Dealers (NASD).

These services are available to qualified users for relatively moderate monthly fees and can be adapted to almost any application. Although their use does of course require computer programs to be written to process the data, these can usually be provided by in-house personnel or purchased for a one-time fee from outside consultants. Consider how this approach relieves the shortcomings mentioned above.

Because they are purchased directly from the exchange, the data are delivered to the user as quickly as possible with no relay through other facilities. Once the data are captured, the user can process them as he wishes: for display, computation, database maintenance, or any other purpose he desires. Because equipment and site configuration are totally under in-house direction, the degree of fault tolerance is up to the user. Systems at the exchanges typically are highly reliable, but phone lines for the data services usually should be run in pairs through independent switching centers.

Cost benefits are a bit harder to assess because they depend on the user's specific needs. For instance, if an in-house computer is already performing the needed computational functions and it can support the interface for the exchange lines, only the line subscription costs and one-time program development costs will be incurred. As a worst case, if no computer system were in place, the trade-off would probably depend on the monthly charges being incurred and the functionality required. Clearly, if the need exists for any degree of data processing or analysis, in most cases a system will have to be installed. Consideration should be given to integrating exchange data lines as part of the effort. If the data are being used for display purposes only (in the manner of Quotron or Bunker-Ramo), the cost effectiveness of a user-owned system will depend on the number of devices or locations that could be eliminated if a system were available. It is generally unre-
alistic, however, to expect that all outside devices can be eliminated.

SYSTEM FUNCTIONS AND REQUIREMENTS

Now that we have established some reasonable justification for considering a user-owned system, let us examine the functions the system will have to perform. The requirements imposed on the selection of the computer system itself are shown in Figure 1. Regardless of the ultimate purpose of the system, at a minimum it should perform data capture and processing, database maintenance, and end-user display functions. Each contributes to the requirements of the system.

Although the content and format of the data transmitted by each source vary from exchange to exchange, most services transmit at speeds from 1200 baud to 9600 baud using either a binary synchronous or an asynchronous protocol. Fortunately, these are fairly common and most types of computer hardware readily support their interface. If not, the average user may find it impossible to read the data lines, so users should be sure to select a compatible machine. Most users are knowledgeable enough to be aware of this requirement, but some overlook the potential pitfall inherent to the “free-wheeling” bi-synch protocol. This must be viewed as a water faucet without a shut-off mechanism; the system must be ready to read the data as fast as they are transmitted, or the data will be lost. It is really a one-way line with no facility for the user to request a halt or a retransmission. Naturally there is a recovery capability. The data are grouped into messages and each message is assigned a sequence number. The user thus can keep track of each message he receives. If any numbers are absent, he can call for a retransmission. This is truly a manual process, with a human placing a telephone call to an operator at the exchange site. This upsets the chronology of the data stream and clearly is to be avoided. Although the retransmission facility is certainly needed (no system is totally foolproof and communications lines occasionally contain noise), the user should select a computer system with sufficient capability and capacity to ensure that messages are not missed during normal operation. Such a system is said to be able to keep up with the lines.

By examining the components of a typical transaction, it is possible to evaluate system requirements both for capability (what needs to be done) and capacity (how much). Each transmission is composed of a minimum of three components. They are data capture, data processing, and database update or computation. The data capture function is probably the most complicated, particularly to those who are unfamiliar with data communications. The less familiar a user is with this technology, the more careful should be the selection of a computer system to support the protocols needed. A system that incorporates vendor-supplied hardware and software to read the data lines on behalf of the user is ideal, because it greatly simplifies the user interface. Systems manufactured by Synapse Computer Corporation and Tandem Computers do this quite nicely, although Synapse conveniently puts more of the processing in its communications controller, thus reducing the overhead at the central processing unit.

Without such features, the user may suddenly find himself writing code to search for control and other framing characters (SYNs, SOHs, and ETXs) and performing parity check computations. The best way to evaluate a prospective vendor’s ability in this area is to request a benchmark to enable evaluation of the level of effort required to create code to read the lines. If you are fortunate enough to have access to the specific exchange services under consideration, you also will be able to assess processing requirements and the number of transactions that can be captured per CPU-second.

Although it probably requires more actual lines of code than any other basic function, processing the data read is surely the most straightforward. Each exchange service provides accurate documentation on message formats. It is the parsing of these messages that we refer to here as data processing. User-specific processing beyond this—and its effect on system requirements—must be treated on a case-by-case basis. If the user facility has no programming personnel, software packages are available for the most common services, or consultants can be hired for customized efforts. Processing power used by this function is also minimal in comparison with that needed by the other components because no input or output operations are performed.

In most systems, the next operation to be completed will be some form of database maintenance, usually to a disk drive or other mass storage device. Two exceptions to this are wholly main memory resident systems or purely event-driven deci-
sion support systems. Because it is not unusual for a database to contain more than 5 megabytes, those systems that are wholly or predominantly memory resident typically monitor only a subset of the securities carried by the line services, or they may use a hybrid memory-disk approach. They rely on data maintained, processed, or retrieved from main memory and do little or none of the I/O operations that bog down systems and consume CPU power.

Great care must be taken with these applications, however, to protect the data in the memory from system malfunctions. It is usually recommended that duplicate copies of the data be kept in the memory of separate processor modules to facilitate recovery from malfunction or data corruption. Unfortunately, this is costly; large amounts of memory, processor modules, and disaster-recovery hardware often are necessary to ensure system reliability and data integrity. Purely event-driven decision support systems sometimes perform no database maintenance at all; they merely act on the data as they are processed and then discard them. Here the system burden depends heavily on the particular application and only by precise computation or benchmark can the hardware requirements, capacity, and response time be estimated.

The vast majority of systems, however, do employ disk-resident databases. Many of them in fact perform several file or record updates for each transaction. Let us assume, for the purposes of our analysis, however, that only one database update is required per transaction and that an update consists of two I/O operations, a read (to retrieve the record of interest) and a write (to revise it on disk). The reader can extrapolate the analysis presented here to a particular situation. Some systems, such as the Synapse, employ sophisticated global memory disk cache schemes to reduce the number of I/O operations. Physical database updates occur at user-defined consistency points rather than one per logical update, as is the case with most other systems. Because this can yield an improvement in database management, it is a significant feature to look for in selecting the hardware. This becomes quite clear when a closer look is taken at the numbers involved.

A single, high-speed line operating at 9600 baud transmits 960 characters per second. Using an example of 1000 characters per second (to make the numbers easier) and an average message length of 40 characters, this yields 25 messages per second. If, as with conventional systems, this translates to 50 physical I/Os per second, the amount of hardware required for each data line becomes prohibitively expensive. That is, if a nominal disk operation requires 30 milliseconds' average access time and about 100 ms of CPU time, 50 transactions will use 1.5 disk-seconds and 5 CPU-seconds. Consequently, we would need at least two logical disk drives (actually, four, to allow for mirroring) and five processor modules for each exchange data line desired (exclusive of other processing requirements). Without such features to improve the I/O efficiency, a user may not only spend money unnecessarily, but also may be unable to configure a system large enough for the application (due to coupling restrictions of multiprocessor architectures) or may outgrow it prematurely (by not being able to keep up with expansion requirements and needed enhancements).

**DESIGN ASPECTS AND SOFTWARE CONSIDERATIONS**

We have reviewed briefly the realm of brokerage applications and discussed some of the features affecting the selection of their computer systems and performance. Let us now direct our attention to some design aspects and other software considerations. Any system of the 1980s must embody the one feature that is the computer industry's equivalent of flag and motherhood: modularity. There are many parallels in logic from one exchange service to another and a high percentage of the code written can be used as a skeleton for other services if a system is modular in design. Modularity improves the development cycle from the standpoint of both time and cost.

One design has proved effective in several applications. The basic process architecture (shown for a single data service in Figure 2) can be replicated for multiple exchange lines as the hardware permits. Exclusive of application-specific code, the system is composed of three different processes interacting in requestor-server relationships to perform communications, transmission, and message level processing.

**Communications Handler**

The communications handler executes all I/O operations to read the raw data from the communications controller. It
provides maintenance of all protocol requirements, does block level format validation and error reporting, and accumulates block level statistics.

**Transmission Handler**

The transmission handler parses blocked data into individual messages. It checks message sequence numbers, validates message parameters, collects transmission level statistics, and reports any errors in message level header format and content.

**Message Processor**

The message processor determines message interest and further processing as required, validates message content, and revises and creates—if necessary—pertinent database items (for pricing, volume, and other security data). It can also relay the data via an interprocess queue to a user-defined process (e.g., a computation process or a process for handling a local area network).

Because (in the Synapse implementation) the actual communications line driver and the majority of the protocol management is accomplished by a separate, dedicated processor external to the CPU, all buffering of transmissions is done by the line driver itself. The communications handler simply posts a read, receives and time-stamps each transmission, and sends it on via an interprocess communications (IPC) facility (actually a message queue) to the transmission handler. In the Synapse system, interprocess message acknowledgment is asynchronous to message dispatch. That is, one process can continually send messages to another without waiting for a reply. The operating system manages the interprocess message queues in memory, automatically mapping them to contiguous disk space as they grow. Some earlier approaches on other hardware required buffering of messages on disk, which increased both physical disk I/O and program complexity.

The segregation of functions between the transmission handler and message processor is probably most significant, however. Because the former deblocks messages and performs all message synchronization and control, the message processor’s only function is to complete the decision logic of data extraction and database access. As such, this process can be “cloned” as necessary to keep up with data reception, forming a many-server-to-one-requestor relationship with the transmission handler.

To maintain consistency between the messages processed by the transmission handler and those the message processors must apply to the database, a separate relational database table is used. Each message processor owns a corresponding row in the table where it stores the sequence number (as sent by the exchange) of the last message processed. The transmission handler distributes messages (via IPCs) to each message processor, based on a modulus of the sequence number. This provides an easily traceable series of messages processed by each message processor. In the event of a systemwide failure, the transmission handler simply scans the sequence number table to determine the last message sent to each message processor. This implementation of sequence number synchronization should be noted both for its simplicity (using standard database procedures) and its flexibility (allowing multiple copies of the database-intensive message processor). Also, although it depends on the vendor’s database system for recovery, this method is as fast as a memory-resident array because the sequence number table is in the global memory disk cache and accessible to all running processes. That is, in the Synapse system, all main memory can be shared among all processor modules.

In this approach, one other process needed to be added before tailoring to the application’s requirements. An error handler was created to receive IPC messages from all other processes. It logs and time-stamps all missing sequence number reports, retransmission requests, and system errors to a designated terminal or hardcopy device, while maintaining a disk file log as well. This relieves the other processes of responsibility for logging errors and provides a central system-monitoring capability. Another important aspect is the benefit inherently derived from Synapse’s relational database management system. All database concurrency considerations, including record locking, deadlock detection, and recovery, are handled by the DBMS and not by the programs. This speeds system development and reduces complexity at the user level.

Time and space do not allow examination of all of the branches a system based on this architecture could follow, but several points should be made regarding the way the design might proceed and some miscellaneous features to seek out in the host system. If the system is to be used solely for local data display, reporting, or other retrieval functions, the only development left for the user is to provide programs for database access. These are usually asynchronous or batch operations in relation to the primary database maintenance function. It should be simply a matter of programming to complete the effort. The ease of development rests with the skills of the programmer and the quality of software tools provided by the computer vendor. Users should therefore carefully evaluate each prospective vendor from this standpoint. Other candidate user applications, either computational (arbitrage or decision support systems) or I/O-oriented (distribution through local area networks), will require more effort to complete, but they are similar enough in the next stage to be considered together. Either can be treated as a fourth-level process below the message processor, using the same IPC capability to pass each transaction to the application. Again, the ability of the host system to execute this function and support a modular design to replicate processes at the user level is essential. If the user keeps in mind these features—and the other suggestions mentioned—in selecting any of the computer systems that are highly reliable and expandable, he can be assured of having the foundation of a successful system.
Panel: Business graphics—the experts predict the future

Chair:
ALAN PALLER, AUI Data Graphics/ISSCO, Arlington, Virginia

Members:
NEIL KLEINMAN, International Data Corporation, Santa Monica, California
ANDERS VINBERG, ISSCO, San Diego, California

The focus of this panel is short- and long-term trends in computer graphics use in management. Equipment trends, including the impact of workstations, is discussed. Trends in software, applications, and management are considered. Specifically, software trends include windows, artificial intelligence, and the new one-button user interface for executive access. Applications trends encompass such topics as production graphics, scientific charting systems, and visual early-warning systems. Finally, management issues include planning for complete visual information systems and for the impact of graphics on the relationship between information system departments and top management.

Panel: Optical and video technologies’ delivery systems

Chair:
D’ELLEN BARDES, Alltech Communications, Denver, Colorado

Members:
M. SCOTT ALBERT, Criterion Venture Partners, Houston, Texas
J. OLIN CAMPBELL, WICAT Systems, Orem, Utah
PETER CROWELL, The Advanced Learning Systems Organization, Blackhawk, Colorado
EDWARD S. ROTHCHILD, Rothchild Consultants, San Francisco, California

Available optical disk and image technologies permit the integration of high-quality pictures with existing text/data/graphics management systems. By providing users access to and manipulation of visual files, innovative archival, merchandising, and management applications are made possible.

Which of the overlapping, cost-effective peripherals or integrated solutions should one select—analog videodisk, optical digital data disk, high color-resolution graphics boards, or compact disks? Industry experts discuss alternative visual information systems’ advantages and disadvantages, industry players, applications, and implementation concerns.
Panel: Computers in manufacturing

Chair:
WILLIAM G. RANKIN, Deere & Company, Moline, Illinois

Members:
ROGER M. BURKHART, Deere & Company, Moline, Illinois
DAVID C. SCOTT, Deere & Company, Moline, Illinois

Deere & Company is recognized as a leader in the application of computer technology to manufacturing. This session includes presentations covering a brief history and background, present activities, and future directions. Applications described range from analytical tools used in planning and designing manufacturing processes to direct control of automated devices on the shop floor. Particular emphasis is given to rapidly advancing computer technology and its integration into manufacturing operations.

Panel: Decision support systems—the end user view

Chair:
EPHRAIM McLEAN, University of California at Los Angeles, Los Angeles, California

Members:
FRED BRACHMAN, Brachman Associates, Pennfield, New York
C. LAWRENCE MEADOR, Massachusetts Institute of Technology, Cambridge, Massachusetts
DAVID NESS, University of Pennsylvania, Philadelphia, Pennsylvania

The participants in this panel session have all had extensive experience in DSS. Ness, who has worked on DSS at the Wharton School, presents observations on the status of the field and where he sees it going. Meader, who has recently completed a major study of several companies using DSS, reports on these results. Finally, Brachman describes his experience with DSS at Kodak and the successful applications they have achieved.
Panel: Current market applications for smart cards, keys, and tags

Chair:
LAWRENCE R. KILTY, The Kilty Company, Bethesda, Maryland

Members:
ROY BRIGHT, Intelsmatique, Paris, France
SAM EPSTEIN, Analytics Communications Systems, Reston, Virginia
BILL FLIES, Datakey, Inc., Burnsville, Minnesota
ROBERT A. KITCHENER, Casio Microcard Corporation, Armonk, New York

This session focuses on actual market applications for smart cards, keys, and tags that are in operation and available to manufacturers and OEMs alike. These portable information devices are being used in a host of areas. The panelists present a current overview of four major application areas:
— The Electronic Dog Tag system: Process control applications; medical and personal history systems
— Communications security applications
— Applications in the credit and debit card industry
— Use in telephone, teletext, and point-of-sale applications

Panel: Security—now and in the future

Chair:
WILLIAM BRAMER, Arthur Andersen & Co., Los Angeles, California

Members:
RUSSEL W. JENKINS, Wells Fargo Bank, N.A., San Francisco, California
MORGAN MORRISON, Glendale Federal Savings and Loan Association, Glendale, California
CLIFFORD A. MORTON, Boise Cascade Corporation, Boise, Idaho

In spite of the wide variety of security management tools and techniques available today, most organizations find that gaps remain in their defense. The dynamics inherent in information technologies ensure that this will always be the case. This session reviews today's security defenses and the frequent shortfalls in trying to achieve their full benefit. The participants propose steps an organization can take now to be better positioned for tomorrow's changes.
Panel: Output devices

Chair:
ALAN SOBEL, Lucitron, Inc., Northbrook, Illinois

Members:
CARL MACHOVER, Machover Associates, White Plains, New York
ROBERT A. MYERS, IBM T. J. Watson Research Center, Yorktown Heights, New York
MARK TABAK, Xerox Electronic Publishing Unit, San Diego, California

The quality of the output from a computer—"soft," like a display, or "hard," like a printout—can have major effects on the way people use the output. CRTs produce most soft copies, but flat panels are beginning to appear in serious numbers. Similarly, the dominant impact types of hard-copy devices are meeting increasing competition from non-impact printers. All four of these general types of output devices are described and projected advances in the technologies discussed.