Business Communication Networks— Putting It All Together

Strategies For The Selection & Integration Of Communication Products

■ INTRODUCTION

Although the data communication user of today has access to product selection and evaluation information on thousands of network products, building a network is more than selecting its elements. This article describes the building of a user network, from setting policy to network management and control, with emphasis on how things go together. Host software, transmission facilities, modems, concentrators, multiplexers, and network control systems are described in the context of a business communication environment.

■ THE PICTURE & THE PROBLEMS

If you're a data communication user, an authority no less than AT&T has surveyed your situation and defined the major problems which you face. In its application for the fully separate unregulated subsidiary later known as Information Systems, AT&T listed what it found to be the four major problems facing communication users today: the application orientation of existing networks, the lack of expandability, the lack of control, and the high capital investment required to support new applications.

Application orientation doesn't sound like much of a problem. After all, aren't networks supposed to be related to an application? Not in today's mode of thinking. Most of the users of data communication evaluate each separately and cost justify it either independent of other uses or incrementally with something which has already been done. This often results in the new network either duplicating parts of an existing one or growing out of it without regard for other uses.

Why do users tend to **grow** application networks? According to AT&T, it is because of traditional project costing and an understandable tendency to take the path of least resistance. It's easy to say that modern communication theory favors the development of a communication network as an application-independent corporate resource, but it is rather difficult to force a user with hundreds of terminals to restructure or convert an existing network because it is tied to a single use such as order entry.

The second user problem was the lack of expandability typical of user networks. This again stems from the tendency to plan for a least-cost current network in an environment where future applications may involve business situations which the current planners are not responsible for nor aware of. When your company plans its national inventory control system are the people studying electronic mail for corporate staff present? If you are a typical user, probably not. Thus, the potential for the expansion of the network to handle electronic mail is likely to be missing, and future planners will call your network **inflexible**.

Lack of network management and control facilities in a network often stems from the reliance on host or front-end processor to provide such capabilities. As networks become more complex and cost pressure mounts, new technologies such as multiplexing may be employed to reduce costs. Since multiplexers create network elements that the host computer cannot control, the user loses the central control of his network from the computer site. **Figure 1** shows the effect of adding a multiplexer/concentrator in a network. The new network is more reliable, but some of the paths are **invisible** to the host computer unless they are switched in. These "phantom lines" may be vital in the strategies of fall-back and recovery, yet the host software has no way of monitoring their state. In addition, newer modems have valuable diagnostic and control features that did not exist when the application was designed and, therefore, are not supported by the computer network control software. While all of these enhancements may be used in existing networks, control of the new facilities may be totally separate from that of the earlier and more conventional routes.

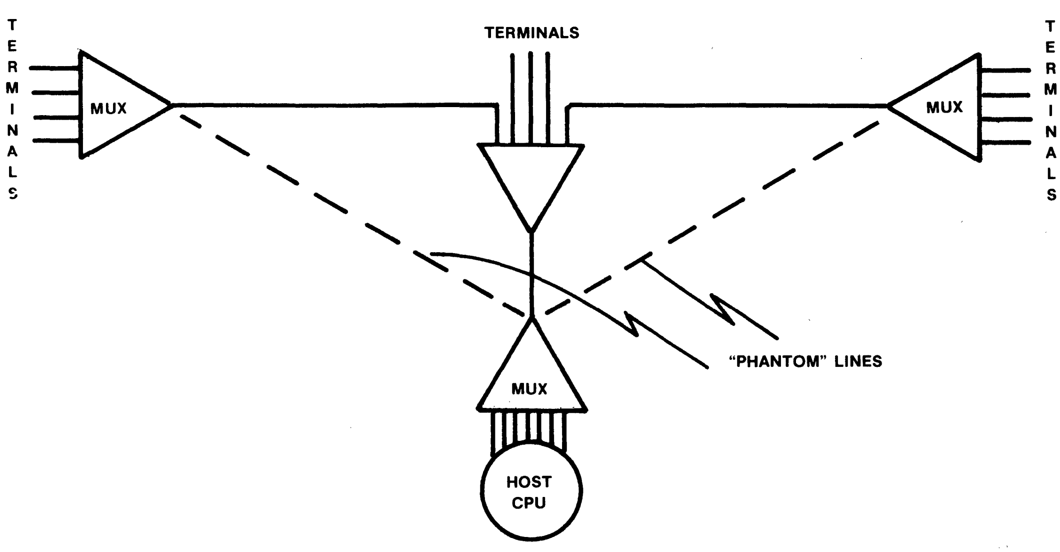


Figure 1 • "phantom lines" hidden from host control.

Business Communication Networks—Putting It All Together

The final problem in AT&T's bag is that of the capital entry threshold for new communication applications. Many users find that good business projects cannot be carried out because the cost of providing the communication facilities is excessive or because the business value of the project cannot be accurately determined. Although this can sometimes be solved by utilizing the facilities of another application, such **piggybacking** is often difficult to sell to the users of the existing facilities, who are concerned about loss of response time, increased failures, and in general of any changes to their comfortable environment.

What are the solutions to these problems? AT&T-IS, of course, says that its Net 1000 service provides them, and to a large extent this is true. Net 1000 is not the only solution, though. The advantage of Net 1000 is that it is a generalized communication resource planned by experts and designed to provide all of the economic and managerial benefits of a network without demanding an excessive capital investment for new users. Any company can provide some or all of these advantages in a private network by careful planning of the network elements. For some, public networks such as Net 1000 may provide the most cost-effective way to communicate. The decision to use public or private facilities is often made very early in the planning cycle, but network problems cannot be solved by looking at the parts of a network, they must be solved as the total network.

■ PLANNING & POLICY IN COMMUNICATION NETWORKS

Data communication is not something that a company uses for its own intrinsic value, it is a tool in the information processing and management policy of the business. One of the first steps in planning communication facilities is thus to define that policy.

You hear and read much about **MIS** or Management Information Systems, office automation, transactional systems, and the like. **Which are you?**

One major element in establishing an **information identity** is classifying the value of information to your business. Some companies use data only as a means of operating; buying, selling, paying, receiving. Data communication may be needed to gather in **transactions** which are the electronic equivalent of paper invoices, checks, etc. Other companies must analyze information in order to do business. Insurance companies, for example, must perform accurate statistical analysis of loss experience in order to set rates. Banks may require an accurate picture of the status of a corporate client in order to approve loans. This type of user may not be able to predict in advance exactly what will be valuable and what will not, and which pieces of data must be correlated in order to support business decisions.

Transactional users place a high value on the timely processing of data since data is a part of their chain of cash receipt and product delivery. Online data entry is popular with users of this type because it improves the clerical productivity, and concepts of office automation or distributed processing are gaining favor here because the structure of information movement is fixed by the applications themselves. A cash receipt may be entered in the accounts receivable system and the general ledger systems but probably not in the inventory system. This fixed flow allows the separation of processing power and its movement closer to the point of transaction—distributed processing. Transactional users, having less reason to gather their data to a central point for processing or correlation, can often employ networks that have no single point of collection—the peer network.

Informational users, on the other hand, view the correlations of pieces of datum as important as their "production" use. They often have very high volumes of data requiring a method of batch entry such as optical scanning or magnetic character recognition, and the cross-matching of information required for generating "ad hoc" reports requires a central storage and processing operation or a distributed intelligence architecture not yet commonly available. These users demand hierarchical networks with remote processing clusters feeding area controllers which in turn feed data to a cluster of host computers.

If the general characteristics of each user type don't help you classify yourself, there are a few key **symptoms**:

- Businesses which obtain a large portion of income from investment of receipts are normally informational in structure.
- Manufacturers are normally transactional users.
- Retail operators with very low margins, such as food stores, are often informational users while those with higher margins are transactional.
- Smaller companies tend to be more transactional in nature while larger ones are more often informational.
- High-technology companies are likely to be more informational.

Once you have identified your classification as a user of information, it should be a key element in your communication planning. Transactional users should generally migrate toward **peer-distributed networks** using host computers and access methods which support real-time transaction processing, the so-called **teleprocessing monitors** such as IBM CICS, for example. Informational users have a need to integrate their data communication and data processing strategies more closely and should investigate **network architectures** such as IBM SNA, Digital Equipment's DECnet, or Honeywell DSA. These will allow a much more flexible use of communication resources.

■ NETWORK ARCHITECTURES & ACCESS METHODS

Most users have little choice in the computer to use for their networks—the one they already have must serve. But within the computer system there may be a choice of communication support packages, and the proper selection here is important because it serves as the basis for the rest of the network. The host software can be divided into the following categories:

- **Access Methods.** These provide a convenient way for the user to interact with the communication hardware. They are normally concerned only with data communication, and provide little or no application or database services.
- **Teleprocessing or Transaction Monitors.** These provide a framework within which a user can develop programs to process single requests for action, or transactions, from a communication line. They include a higher level of communication services that further insulates the application programmer from the technical details of communication lines, and may provide additional services such as database management and screen building.
- **Network Architectures.** These complex packages join user program and communication facilities so that local devices and remote terminals are almost indistinguishable. The application program uses the communication facilities as resources, and the resource is managed by the network architecture itself, independent of its users.

Access methods generally require a program to use the communication line as a device, reading and writing to it to communicate with the users it serves. While the program is loaded, all data from the line is routed there. More sophisticated systems will allow the control of a device by a program, allowing other devices on the same line to be handled by other programs. In monitors or network architectures, the communication software often allows the user to enter a series of requests or commands that are dynamically routed to the particular program possessing them. A single terminal can thus update an accounting file, read the status of an order, and change the address of an employee without requiring a central computer operator to switch the device from one computer, computer port, or program to another.

Selection of host software support requires that you as a user evaluate not only your needs but also your capabilities. Most vendors offer access methods in varying levels of complexity, and the programming environment differs dramatically between a simple access method and an advanced network architecture. In general, the network architectures allow less sophisticated application programmers to work in a communication environment, but will cost more in direct software fees and in configuration of the more complex hardware normally required. On the other hand, the "simple" access method may cost very little, run in a minimum configuration, and be nearly impossible for your staff to learn to use.

If you are a small user, probably transactional in your use of data communication, you will probably find a teleprocessing monitor

Business Communication Networks—Putting It All Together

or transaction package the best choice. These packages, such as IBM's CICS or Informatics' TAPS, will provide your programmers with a consistent communication environment within which applications can be developed. The selection of a specific package should be made on the basis of the features of the monitor, particularly the ease with which it adapts to various protocols. A monitor that handles IBM binary synchronous communication will not serve a user with plans to employ another protocol such as SNA.

The following features should be evaluated in TP monitors:

• **Is it limited to a single vendor, single operating system, or a single access method?** Such limitations are common in the packages provided by hardware vendors, but restrict the growth path of the user. If you are certain you won't change vendors or operating environments, restrictive packages may provide all you need at lower cost.

• **Will it handle the protocols in use or planned?** Many TP monitors use a host access method as a basis and have no restrictions in protocol not inherent in the computers, while others do not support all protocols.

• **Is database support included?** The more complete the operating environment, the more easily it is used by relatively inexperienced programmers. Real-time communication applications are more than just line handling, and if your programmers cannot handle the data communication part they may not be able to handle the real-time data management either. Be particularly careful of file and record lock facilities to prevent collision in accessing the same files from multiple users.

• **Is there a screen builder program?** The design and development of good screen formats for data entry is a major task and has a significant impact on the productivity of the users of the system. Some packages allow programmers to develop a screen from a set of simple application-oriented definitions, and this saves considerable time and debugging.

• **Is there a query language?** The ability to retrieve ad hoc reports from the database is a benefit, particularly to informational users. Transaction users can use the query language or "bread and butter" reports like transaction registers and daily activity logs, saving programming efforts for more important tasks.

• **Are there provisions for security against unauthorized access?** Password protection keeps a stranger from accessing the system, but not necessarily an authorized user from seeing sensitive data in another area. File and program level security or closed user groups are valuable in keeping the wrong people out of your files. If the TP software doesn't provide it, you'll probably end up writing it yourself.

Users with more sophisticated needs will probably find that a total network architecture will serve them best. These extend the services of TP monitors to what can be called the "logical device" concept, where a program can operate with a local user, a remote user, or a data file and not need to know which is which. There are limitations to this, of course, but most of the mainframe and minicomputer vendors are working to provide such facilities for their systems. The international standards work on protocols such as X.25 has advanced to the point where truly user-independent interactions, as defined by the OSI Reference Model, are now possible. Vendors are busily at work developing OSI products, although none are yet available and probably will not be available until 1985.

There are some key points to consider in the selection of a network architecture:

• **Equipment supported.** Since most architectures are still vendor specific it is important to be sure that the equipment which the vendor supplies is capable of handling the entire communication and user application, and that the cost is reasonable. Remember, a network architecture also establishes your operational environment for business programs.

• **Device support.** Will the architecture support the devices you plan to use or already have? Will there be any restrictions in the features provided by the architecture with off-brand or unusual protocols?

• **Network structure.** What type of connection structure does the architecture expect? If terminals cannot directly enter the host computer, the number of communication and cluster controllers may impact costs significantly, especially if the number of terminals at any one location is relatively small. If the architecture does not support host-to-host connections easily, it may be unsuitable for many multihost users.

• **Network management.** Since the users are almost completely isolated from the communication facility, the network architecture must manage this itself. Are the reporting and diagnostic facilities of the system adequate? Can spare lines or alternate routes be utilized? Is it easy to place redundant equipment in the network and switch to it? External network or technical control is generally possible even with network architectures, but non-integrated control is often difficult to coordinate. How do you tell the network architecture's facilities manager that you've switched a line? What do you have to tell the users, if anything?

• **Design direction.** Nearly everything in data communication is evolving toward something, including your own operation and the architecture you select. If they are not going the same place, an eventual migration to another system is required. **With network architectures, you do not change concepts easily.** Explore the ultimate goals of the architecture before committing to it, because the application programs you develop under it will probably be much less transportable than non-communication programs that perform similar functions.

What about access methods? Why leave them to last? Most access methods are primitive-level interfaces to the communication hardware which require an experienced data communication programmer to use. If your environment is such that this type of individual is likely to be available, access method programming may be a reasonable choice. Unless your company is large, it probably is not. But if you decide to use access methods directly, don't let the fact that they are relatively primitive lead you to expect that they are also very flexible.

Access methods can be broadly classified as direct or queued. Direct structures require that the user program issue "reads" and "writes" directly to the communication line, a practice which increases programming complexity and tends to dedicate a given line to a single application. In queued structures, the access method manages the line on your behalf and delivers messages from or to it when requested by the programmer. A "read" or "write" command in such a system actually reads from or writes to a queue of messages. Some systems permit the queueing of data to a disk device if the application program that must receive it is not loaded or temporarily unable to accept data.

Some computer systems bundle access methods and host operating systems together, or offer several operating systems which in turn are supported by several access methods. It's normally best to view these combined products as combined products rather than trying to separate them. Catalog the languages available in each system, the number and type of special-purpose facilities like database management, and the communication implications of each. Don't forget to consider any restrictions on multiprogramming or multitasking which their operating system imposes. In particular, check the number of tasks that can be run simultaneously and the relationship between the number of tasks and the number of lines or terminals.

A final point on the evaluation checklist for any communication software is its ability to adapt to various transmission media. In many of the complex network architectures, access to public data networks or satellite links can be transparent to the user, while others will either not perform with specialized transmission systems or will perform with an unacceptable level of degradation.

■ ANALOG NETWORKS—THE COMMON CARRIERS

Most data communication is still maintained over dedicated or switched telephone circuits, and very few users can plan a network that totally excludes these paths. The term **analog** is applied to these lines because they carry frequencies in the audio range, such as the human voice. Analog lines can be classified according to the range of frequencies they can carry; sub-voice grade cannot carry the human voice properly, voice grade carries the frequencies required for normal conversation. The total span of frequencies carried by a circuit is called its **bandwidth**.

Business Communication Networks—Putting It All Together

Since computers and terminal devices used in data communication output digital data rather than audio or analog data, the output must be converted to analog tones through use of a **modem**. The type of conversion, frequencies used, and various technical features of the modem determine the maximum speed which the modem link can support. Modem speed is restricted by the bandwidth of the channel (the greater the range of frequencies available, the higher the speed) and the noise level on the line.

There are many kinds of modems, and the points of differentiation have to be examined by prospective users or there is a high risk of improper selection. To do this, we must introduce some terminology; half- and full-duplex, dial and leased lines, synchronous and asynchronous, and point-to-point or multipoint.

Analog circuits may be operated as two-way alternate paths, called **half-duplex** or two-way simultaneous paths, called **full-duplex** as shown in **Figure 2**. Half-duplex operation allows data transmission in either direction, but only in one direction at a time. With full-duplex, data can flow in both directions at once. Half-duplex connections are normally sufficient for terminal-to-computer interactions because human operators tend to either read the terminal data or key their own but not both at once. Such read-change-send interactions are sometimes called **logically half-duplex** because simultaneous communication is contrary to the interaction of the two stations regardless of the capabilities of the path.

Full-duplex communication is often required for computer-to-computer exchanges because the high volume of information moved makes it undesirable to restrict a station to either sending only or receiving only, even for a short time. While half-duplex data paths can be supported by a standard dial-up connection (sometimes called **two-wire** because of the number of voice-carrying wires in the circuit), full-duplex communication can be handled over such paths only at low-to-medium speeds, currently, to 4800 bits per second (bps). This is because the bandwidth restrictions on data capacity of the line apply to the total of the data sent at one time in either direction. Full-duplex operation above 4800 bps requires two circuits, one in each direction.

Another characteristic of analog communication is the way in which characters are sent on the circuit. For low-speed applications, each character can be marked with a **start bit** and a **stop bit** and sent whenever it is available. This method is called **asynchronous** because the sender and the receiver do not have to exchange precise information on when data is being sent, called **clocking**. But async, as it is normally called, has a high overhead because of the need for start and stop bits, and at very high speeds it is difficult to keep the modems at each end of a path properly tracking with no data or data in random spurts. A solution to this is to exchange clocking information between the modems so that each station knows exactly when the other is putting data on the line. This is called **synchronous** communication, and nearly all high-speed transfers over analog circuits are synchronous.

Analog circuits can be leased from the phone company or accessed through direct distance dialing (DDD). The relative economics of the two methods depends on the amount of service you need and the distances involved. Here are some rules of thumb in evaluating leased lines versus dialed lines:

- Infrequent connections (less than several hours per day total call time) are almost always more economically handled by dial-up service or a high-volume user program such as WATS (Wide-Area Telephone Service).
- High data rates (over 4800 bps) will nearly always require leased lines.

A type of circuit that always requires leased lines is called **multipoint or multidrop**. This is a phone line that follows a path with "stops" or "drops" along the way, to which terminals or other devices are connected. The line, being shared between users rather than dedicated to a single user, is normally less expensive than point-to-point service for all the stations, but the management of multidrop service is more complex, and there are obviously limitations on the response because of competition for the line. **Figure 3** shows the advantages of multipoint or multidrop lines.

Selecting a combination of these features for your particular environment isn't as complex as it might appear. Here are a few basic rules:

- Check with the equipment vendor first. There may be a recommended (or even required) set of modem/line options for your system. At least such a check will normally reduce the alternatives list.
- Slow terminal devices, such as printers, normally operate on asynchronous circuits at speeds of 1200 bps or less. A printer without a keyboard may operate satisfactorily at half-duplex, but most keyboard devices expect the computer to **echo** the character or it will not print locally. This requires a full-duplex modem.
- High-speed devices such as computers, or terminal systems such as IBM's 3270, require synchronous modems. If the operating speed is over 4800 bps, the modem manufacturer may recommend leased, **conditioned**, lines. The conditioning process assures that the line's performance will be within a tighter tolerance than normal lines, and separate charges are assessed for conditioning depending on the type and the performance desired. If you're selecting a modem, be sure to check the conditioning requirements carefully; it may be more expensive in the short term to buy a modem with high-quality equalization logic that eliminates the need for conditioning, but it may pay for itself in the long term.
- Most multipoint lines and many point-to-point lines are **polled**, which means that a master station must send a request for data to each slave station attached at frequent intervals (several times per second is not uncommon). If half-duplex lines are used, the master modem must **train** to each new slave, and the slave answer requires that the direction of transmission be changed. This process is called **turning the line around**. There is a built-in delay in modems called **clear-to-send delay** which holds up the sending station for a time period to allow the modem to train. This delay is often quite long, approaching a quarter of a second. To

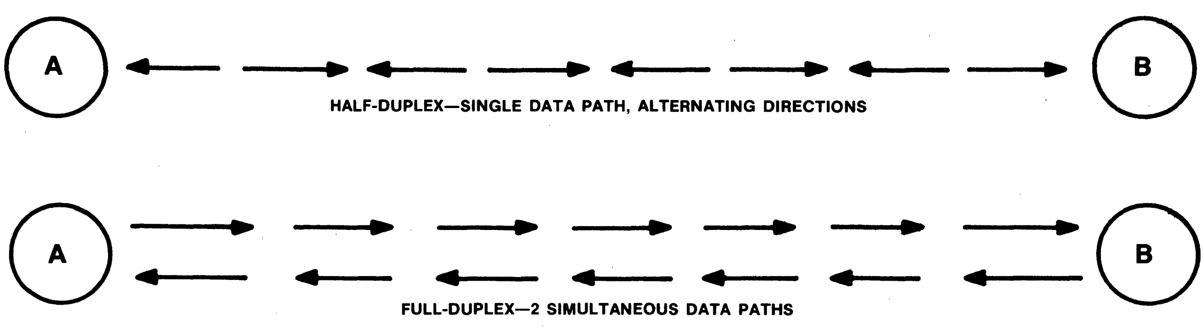


Figure 2 • full- and half-duplex transmission.

Business Communication Networks—Putting It All Together

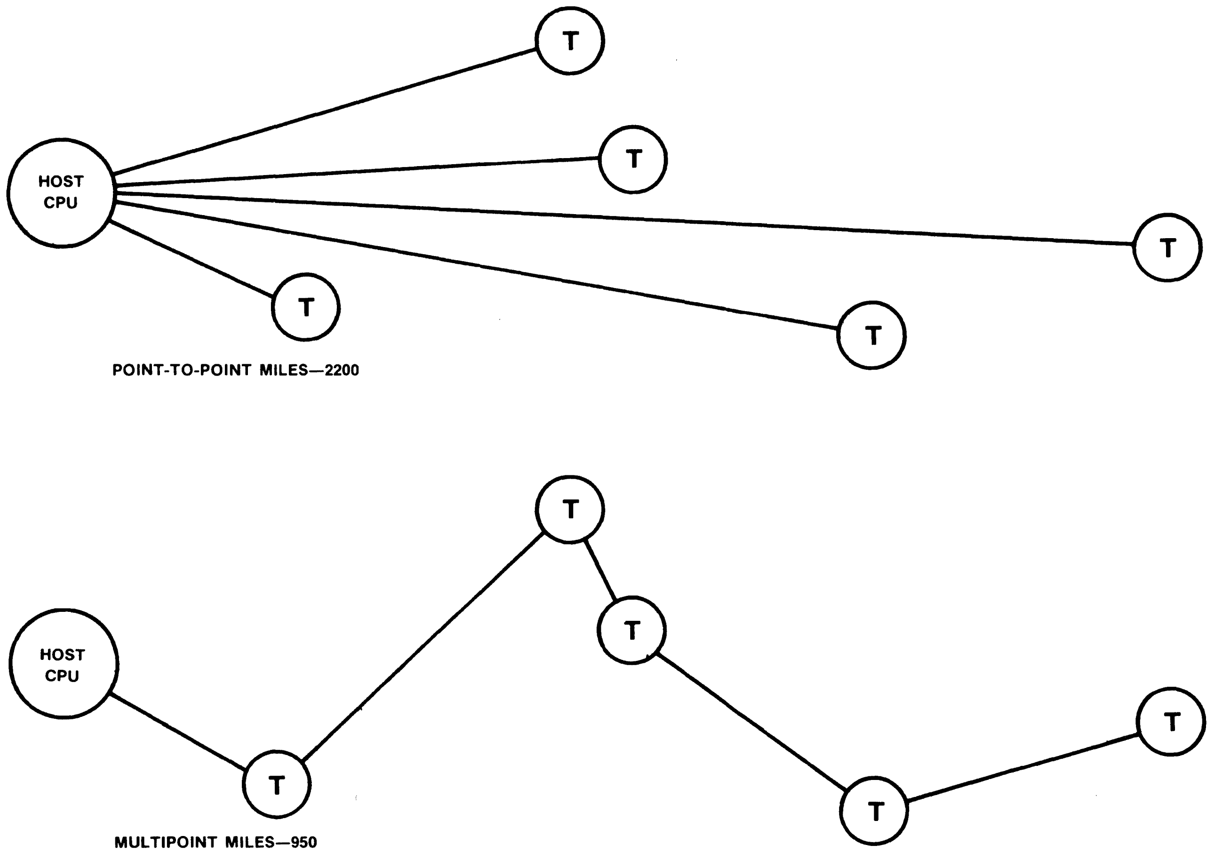


Figure 3 • multipoint line application.

prevent the delay from slowing the polling of the line, special modems called **fast-poll modems** are available which train quickly. Consider these if you have a line with many stations.

• Modem or turnaround delay also affects data capacity of a line. A quarter of a second delay is an entire block time to a batch terminal operating at 9600 bps. When long sequences of data are to be sent, for applications such as remote printing or RJE, it is almost always wise to consider full-duplex service. This will eliminate the turnaround delay and dramatically cut the time required to complete a big print job. This rule applies only to high-performance lines that use error-correcting protocols such as IBM bisync.

• Most users will need to buy modems in pairs, one for each end of the circuit. If one station already has a modem, as would be the case if you were connecting to a timesharing service or service bureau, the modem you purchase **must be compatible** with the existing one. There are two main standards of compatibility, AT&T-IS and CCITT. Many vendors advertise as **103/113 compatible** or **AT&T-IS 209 compatible**. There is a high probability that such units are compatible with AT&T-IS modems, but a lesser chance that they are compatible with each other. The CCITT, an international standards organization, defines modem standards for speeds from 200 bps to 9600 bps, and modems that comply with one of these standards are normally compatible with others that do. It's always a good idea to try any combination of unlike modems first, and as a general rule **if you have to buy two, buy two of the same**.

■ PRIVATE NETWORK DEVICES—MULTIPLEXERS & CONCENTRATORS

Users with heavy communication loads, large numbers of lines, or unusually high reliability requirements may find that even leased line networks are not adequate either from a cost or performance perspective. Point-to-point and multipoint networks may seem to duplicate circuits when many terminals are scattered in a relatively small area but none are co-located. This duplication cannot be avoided within the constraints of point-to-point or multipoint architectures, so a new network structure must be proposed—the nodal network.

Most users tend to think of networks as having the classical **star** structure shown in **Figure 4**. This may be the best network design for a few widely separated devices, but better line economy can almost always be achieved by introducing some points of concentration in the network. Network architectures such as SNA, DECnet, or packet networks achieve this by network elements called **nodes**. These nodes are an active part of the network and cooperate with the host computer or computers to move information efficiently and recover from line failures through alternate routing.

Users without a network architecture can gain some of the advantages of nodal networks by using devices such as multiplexers, network processors, or concentrators. The distinction between these devices has become blurred as vendors offer new features on each, so some definitions of the terms must be provided:

• **Multiplexers** are applied in pairs to collect a series of lines into a single line at one end and fan them out to the same number at the other. Multiplexers do not change the protocol or the number of paths, so their use is largely transparent to the user's application and equipment. Although multiplexers may provide additional services such as switching to backup facilities or alternate routes, the primary purpose of multiplexers is to reduce line costs by combining many relatively lightly utilized paths into a single, more heavily utilized path, thereby eliminating the cost

Business Communication Networks—Putting It All Together

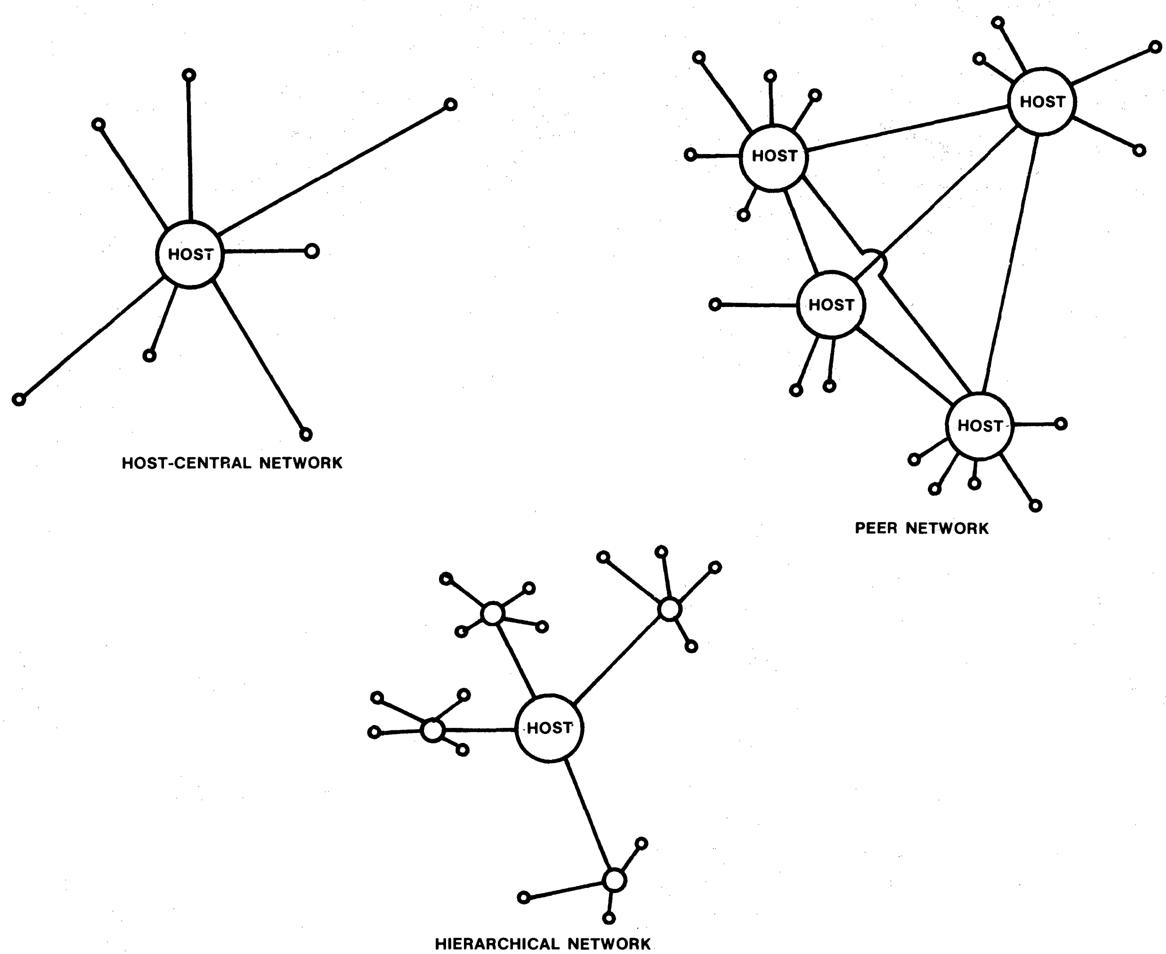


Figure 4 • central, hierarchical, and peer networks.

of additional lines.

• **Concentrators** collect multiple lines into a single line, but do not pair with a partner to split them out again. If the protocol of the concentrated devices permits multiple users on a line, the effect may be only to reduce the number of ports into a computer at the end of the concentrated link. If not, the concentrator must use a multidevice protocol on the concentrated line, and the user at the other end must support that protocol rather than the one used by the individual devices being concentrated.

• **Network processors** are multiplexers or concentrators that add features such as routing data to multiple "trunk" lines and/or protocol conversion. Lines may be run from one network processor to another to provide alternate paths in case of line failure or to balance communication loads. Network processors are more or less transparent to the user, depending on the features selected.

Multiplexers are probably the most popular communication device apart from modems. **Figure 5** shows the effect of placing multiplexers in a user network. The remote stations are grouped into cluster points where a multiplexer collects them and moves them over a single line to the central site. Here other multiplexers restore the original number of lines so that the computer configuration does not change. The effect is a reduction in the number of lines, and therefore, in the cost of the network. As the cost of lines increases and that of multiplexers decreases, the circumstances that justify their use become broader. Some users can actually save money by multiplexing as few as two lines.

There are several types of multiplexers that differ according to the technique used to share the common line. The oldest scheme, called **frequency-division multiplexing (FDM)**, separates users by assigning them different frequencies within the analog bandwidth. This scheme is much too restrictive and is seldom used today. Another method, called **time-division multiplexing (TDM)**, gives each user a time slice on the common line. This slice, normally the space required by a single character, is permanently assigned to the user, and goes empty if the user has no data to fill it when it is offered. The fixed nature of the allocation for both TDM and FDM is unappealing when the use of the lines to be multiplexed is very light. Since typical utilization of terminal links is 10 to 20 percent, both TDM and FDM waste line capacity by granting it to terminals that statistically are unlikely to need it. Fixed allocation of line capacity, such as TDM or FDM provides, is most useful when the lines being collected are already heavily utilized. In this case the cost savings is achieved by taking advantage of the fact that a single circuit of high data capacity (9600 bps, for example) is not twice the cost of two circuits at half that speed. Small TDM devices called **bandsplitters** are often used to combine several slower lines into a single high-speed line. Bandsplitters are usually combined with modems for maximum economy, and may be useful where up to four low-speed circuits with high utilization or stringent response requirements may be combined into a single trunk.

A more efficient concept for multiplexing now used by most

Business Communication Networks—Putting It All Together

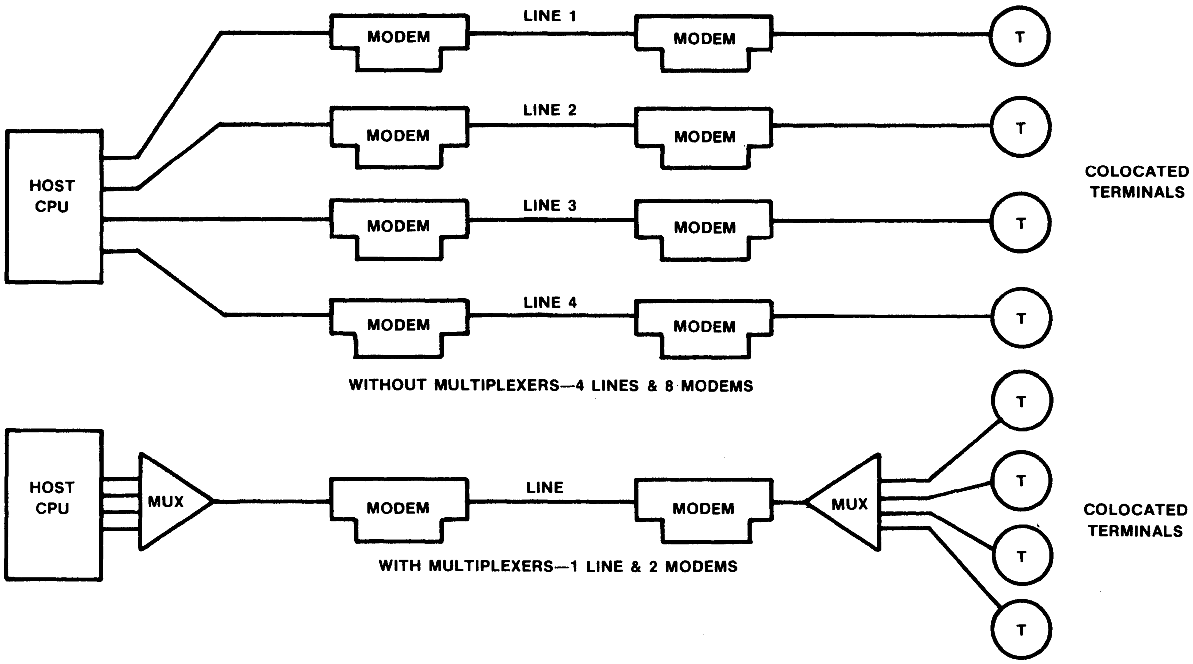


Figure 5 • application of multiplexers.

vendors is called **statistical multiplexing**. This system collects data from the incoming lines and inserts the data **and an identifier of the source line** onto the common trunk. The opposite multiplexer then extracts the data and source ID and gives the data to the equivalent line on the other side. If a line presents no data, it is not allocated space on the high-speed link and that space can be used by other stations. **Figure 6** contrasts the FDM, TDM, and statistical techniques.

The main advantage of a statistical multiplexer, or **stat mux** is easily illustrated. In TDM or FDM techniques, each incoming line must be allocated enough capacity to handle its maximum rate of data delivery. For example, a TDM could combine eight 1200-bps lines onto a single 9600-bps line, but not 10 or 12. This is because the TDM has no way to utilize the empty slots of one user with data from another. In a stat mux, the limit on the number of lines that can be combined is the **actual** data traffic, not the potential traffic. If your 1200-bps lines are actually utilized at 20 percent, then each only requires 240-bps of high-speed line capacity. Great! That means 40 lines over a 9600-bps link! Not exactly. The average data rate over the lines may be only 240 bps, but the data will not be evenly distributed. If the data on all lines arrives at random intervals, there will be times when the number of characters waiting will exceed the capacity of the common line. This causes a "queueing delay." There are a series of formulae to predict the queueing delay for various conditions, and all relate the delay to the **average utilization** of the common high-speed link. If the delays are plotted on a graph, a curve of the shape shown in **Figure 7** results. As you can see, when utilizations reach about 70 percent the delay climbs rapidly. In our previous example of 1200-bps terminals at 20 percent utilization, we would reach this 70 percent high-speed link utilization at about 28 devices, still a considerable improvement over the eight allowed by TDM.

If you would like to calculate the limit of the number of lines a statistical multiplexer could combine, follow these steps:

1. Your high-speed line is probably synchronous, so its speed divided by eight is the data rate in characters per second. For a 9600-bps line, this is 1200 cps. Since there is some overhead in statistical multiplexing, use 1000 cps as a better guide. Now, if we want to stay off the sharp climb of the delay curve, we must figure 70 percent of that, or 700 cps is our target utilization.

2. Your input synchronous lines also use eight bits per second for each character per second, so divide each of their speeds by eight to get their capacity. Now multiply that figure by their utilization. If you don't know the value, assume that 3270-type lines are 40 percent utilized and 2780/3780 batch/RJE lines are 60 percent utilized. Add up all the results.

3. Your asynchronous lines use 10 bps per cps, so divide their speed by 10. You can assume that they are 20 percent utilized unless you have better figures. Add these numbers up, too.

4. Now, add the total of the synchronous and asynchronous lines. If the value exceeds the high-speed line capacity (700 cps in our example of 9600-bps trunk lines), the delay is likely to be noticeable and perhaps prohibitive.

For example, if you have four asynchronous lines at 2400 bps running IBM 3270 terminals and 10 asynchronous lines at 1200 bps, your calculation would look like this:

1. High-speed line capacity is 700 cps.

2. Synchronous load per line is 2400 divided by 8 or 300 cps at 40 percent utilization, or 120 cps. Times 4 lines, this is 480 cps.

3. Asynchronous load per line is 1200 divided by 10 or 120 cps. At 20 percent utilization, that's 24 cps per line or 240 cps for all 10.

4. Adding the sync and async load, 480 plus 240 is 720 cps, which is slightly over our target of 700. When utilization is this high, it's a good idea to check the utilization figures carefully or to drop some of the lines out of the multiplexer.

Multiplexer delay is not a popular issue with vendors or users. Most manufacturers state that their delay is relatively short, one or two character times. This would mean that the delay of a character at 1200 bps would be about eight one-thousandths of a second, or eight milliseconds, a figure users could easily accept. Anyone who has ever measured multiplexer delay finds that the manufacturer's figures are inaccurate. For the older style of fixed-allocation multiplexers such as TDMs or FDMs, the manufacturer's figures are accurate, but for statistical multiplexers under what would be considered normal loads,

Business Communication Networks—Putting It All Together

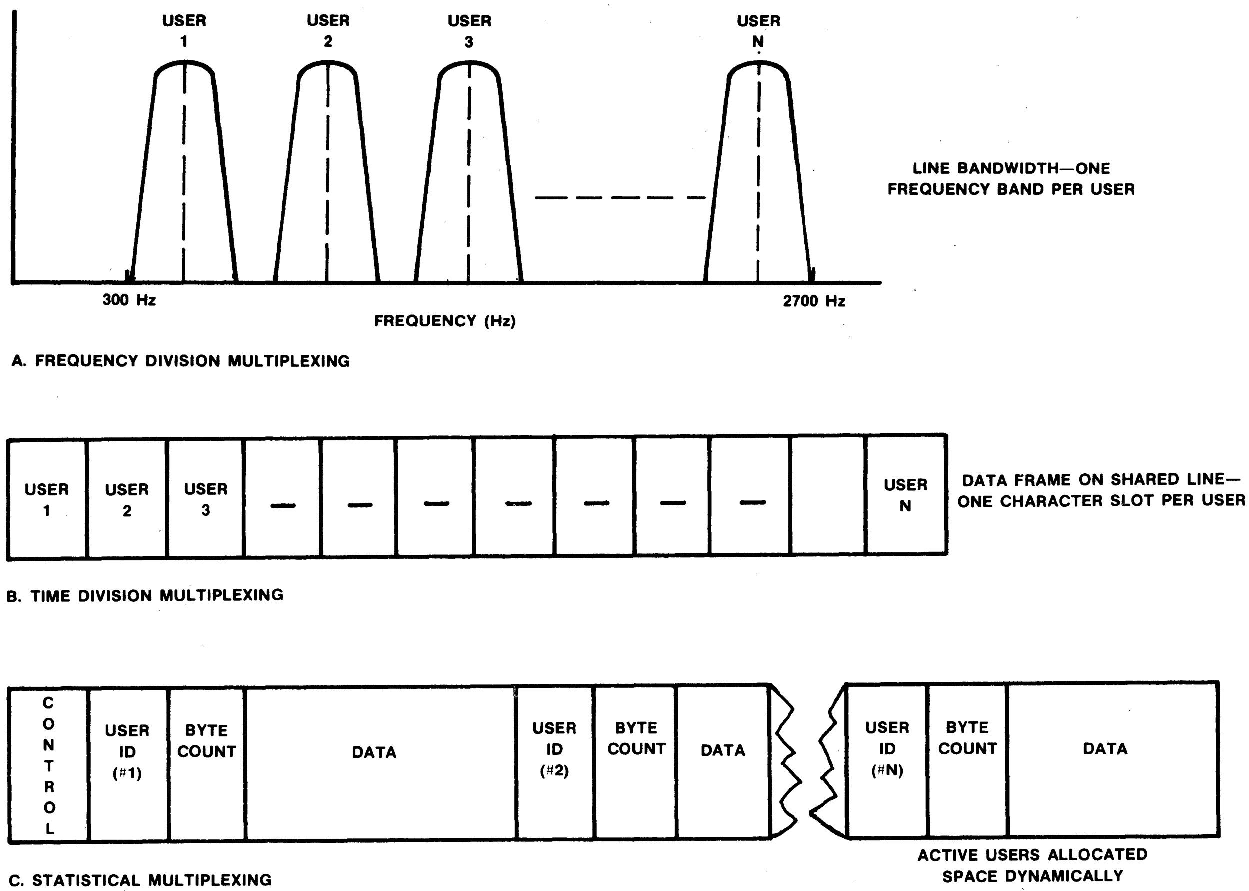


Figure 6 • multiplexing techniques.

delays are normally greater than advertised, and increase with loading as shown in **Figure 7**. Vendors, rather than deal with the delay issue, prefer to relate statistical multiplexer performance to the ratio of the total input speeds to the link speed, a measure called **compaction ratio** or **overbooking ratio**. While these published ratios are useful as a guide, they do not serve as a point of vendor selection because the design differences in the units tend to become unimportant in their contribution to delay at higher levels of utilization, where all units of all vendors approach the statistical curve already described.

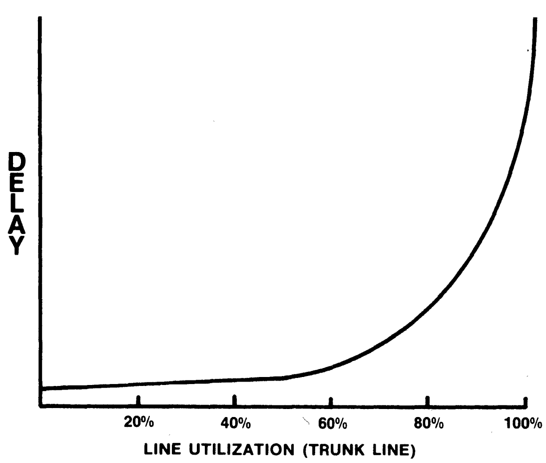


Figure 7 • statistical multiplexer delay.

Concentrators, once relatively popular devices, have fallen into a decline because of the problems in finding a suitable concentration protocol. Because concentrators do not operate in pairs to restore the original line configuration at the host in the way multiplexers do, the host computer must process the concentrated protocol directly. Recent interest in the international packet switching protocol, X.25, has led to the availability of concentrators which use X.25 as their **output protocol** and either asynchronous or binary synchronous (BSC) as their inputs. These units, called **PADs or BPADs** for "packet assembly/disassembly" facilities, provide many of the advantages of multiplexing at a lower cost **if the host computer has X.25 support**.

Buying and using devices like multiplexers or concentrators is often difficult for first-time users because the network concepts they employ are not familiar. Here are some guidelines in evaluating your application for private network devices:

• Evaluate your traffic pattern as the first step in the selection process. Networks are normally built for reasons of line economics, and lines that are either already fully used or that serve isolated areas are poor candidates for combining into networks. If you have lines that are less than 50 percent utilized, that operate under 4800 bps, and that serve users who could be grouped into geographic clusters you should consider private networks for cost savings.

Business Communication Networks—Putting It All Together

• Configure your network to take advantage of natural points of concentration in locations where equipment already exists. **Figure 8** shows a user communication network with the new lines for application of concentration/multiplexing shown as dotted paths. While the **best** point to concentrate a cluster of users is its center, for minimum line charges, it is uneconomical to build an office at a new location merely to serve as the location of the concentrator. If your business is service-oriented and you have lines to customer locations, finding a point of concentration will be a special challenge. Customers may object to having lines to other users terminate at their location and the other users may fear security breaches if the data is proprietary. Be sure to cover these issues, as well as insurance of and access to your equipment, before making a decision.

• Some communication protocols, especially those used by electronic funds transfer equipment, may have very short **timeout** periods. This means that the host system expects the terminal to respond to its inquiries or commands very quickly. Any form of resource sharing used to build networks may introduce **unacceptable delays** and may cause the devices to fail to operate. Polled asynchronous protocols designed along the lines of the Burroughs Poll-Select or NCR standards **may** have these short timeouts. **Avoid using any form of resource sharing if the timeout interval is less than one second, and try to benchmark the products in your environment if timeouts are less than three seconds.**

• Some multiplexer and concentrator vendors limit the aggregate data rates of all the input lines. For example, a vendor may say that the multiplexer has 16 channels of input with an aggregate limit of 76,800 bps. This means that no more than 16 lines can be multiplexed with that device, and that the total line speed of all lines can not exceed 76,800 bps. Since that is an average of only 4800 bps per line, users with requirements for higher speeds might find this unit unacceptable. **Don't make a selection on this basis alone.** Most users report that the ratio of the total input data rate to the rate of the high-speed multiplexed link should be less than 8 or 10 to 1, even on lightly used lines. Performance may limit your application before vendor-enforced restrictions apply.

• Some special devices used in data communication require a nearly complete set of control leads from the device interface, normally the 25-pin RS-232C standard. Most multiplexers and concentrators will handle only two to five control leads, so devices that need more may not function. The secondary channel leads are almost never available, so if you need such facilities you must **seek alternative solutions** to multiplexing/concentration. Most users will not find this a problem.

• Some network products perform protocol conversion so that unlike devices can communicate. While this can be a benefit to multidevice users, **it must be evaluated carefully**. Just stripping off the IBM bisync control characters from the start and end of a message will not make the data within compatible with a DEC terminal. Protocol conversion within network products is most useful in permitting host-to-host connections where data presentation formats can be negotiated by the programmers involved.

• **Don't neglect the TDM-type device** for applications where the lines to be combined are 4800 bps or less and highly utilized, or where wideband (56K bps) service is used. A TDM is less expensive than a statistical multiplexer or concentrator and offers less delay. Use it where input line utilization is so high that the statistical sharing of the resources is improbable.

Once you have settled on an application and general type of product, look for the following characteristics in each vendor offering:

• **Cost.** Most users buy multiplexers and concentrators to save in-line costs. Don't let feature shopping lead you away from the primary goal.

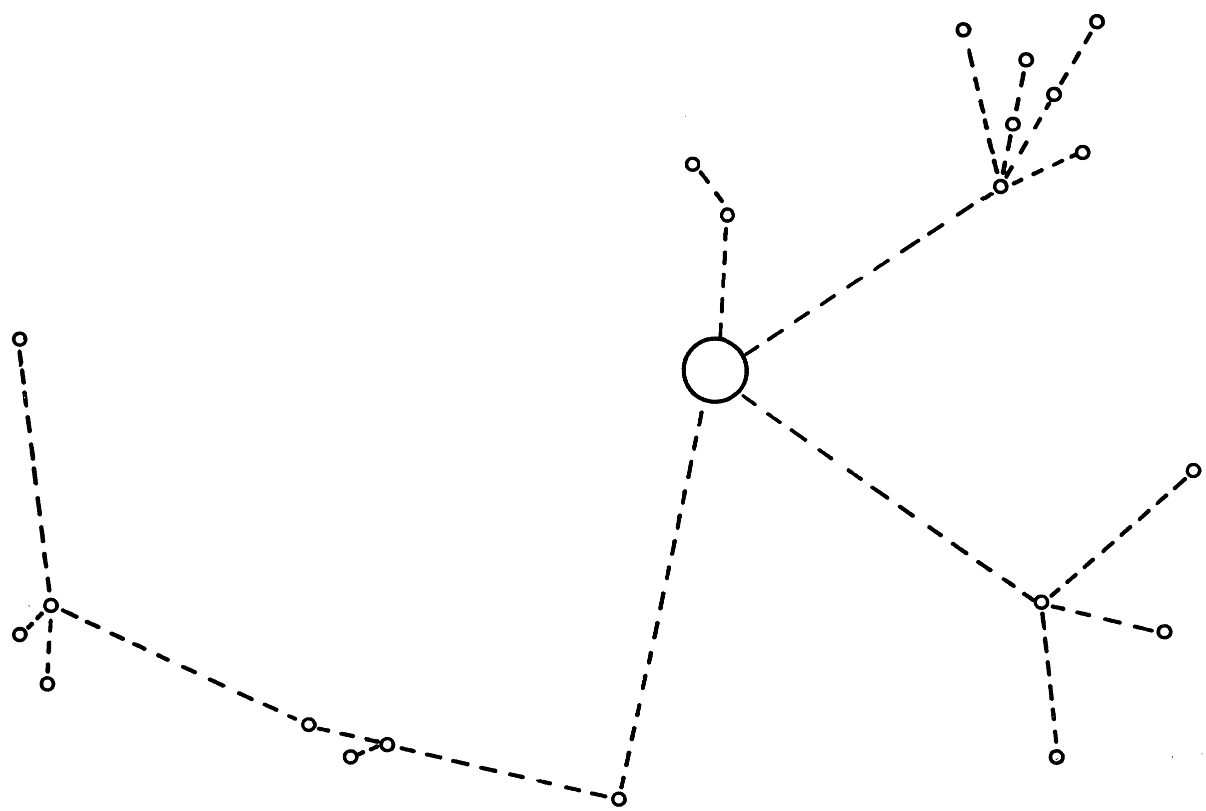


Figure 8 • building a network.

Business Communication Networks—Putting It All Together

• **Protocol support.** Be sure that the devices you select support all the protocols you intend to use with it. If you have any defined expansion strategies, be sure that the devices either support them or will not take part in the expansion. In some cases it is justified to buy equipment that will be obsoleted in a few years if payback has been reached, but try to find something you can continue to use even after network changes occur. Also, be sure that the multiplexer/concentrator can handle the problem of **echo** for asynchronous characters. Most async applications use a host echo of the character to cause the keyed character to display to the user. The delays inherent in multiplexing may require an echo at the multiplexer to maintain operator keying speed. You must **disable your CPU echo in this case** or characters will display twice.

• **Flow control.** Asynchronous terminals and printers may respond to requests to pause in sending data if the rate of data arrival is temporarily too high for the CPU to buffer. This function can be used by the multiplexer to prevent loss of data when many users try to send at the same time, choking the shared link. If you want to send data to a printer, it is almost always a good idea to have flow control in the multiplexer because there is no "conversational" interaction that would naturally cause the system to stop sending after a short message. With keyboard terminals, flow control is useful where the messages sent by the CPU are very long. The value of flow control increases as the ratio of the input data rates to the high-speed link rate increases. **If you are concentrating many terminals that are lightly utilized, you probably need it.** Otherwise a competition for the shared resource which results from, for example, everyone starting at the same time in the morning causes the multiplexer or concentrator to overflow its internal storage and data is lost.

• **High-speed line characteristics.** It is highly desirable that the devices you select operate on a wide variety of transmission paths. Be sure that the protocol used on the concentrated line is relatively immune to satellite delay, and that high-speed operation is supported. Almost all these devices provide for operation to 9600 bps, but some provide standard support for 19.2K bps, and some will provide a special feature to run at speeds up to 72K bps. If the cost of expansion is not prohibitive, the confidence such growth potential can provide is considerable in an environment where usage is tending to increase.

• **Network management.** Private network elements are often transparent to the user and thus to the existing host facilities for management of the network. Be sure that the devices you select have the facilities for testing both themselves and their shared data path, and that the tests can be conducted from a central point and with a high degree of user-friendliness (ease of operation). For small users, built-in test and configuration controls are the least expensive, but larger systems operate best with a terminal device as a supervisory console. If you buy a unit with support for more than eight lines you should have at least the option for supervisory terminal support.

• **Upward mobility.** Most users initially install networks that are "star" point-to-point in nature. When the first concentration devices are placed in this environment they tend to preserve the initial star structure, but as the user grows, the network may evolve into a truly "nodal" structure. **Figure 4** shows a network evolution. Unless you are sure that your network will not grow to a nodal structure with alternate routes and fallback devices, you should be sure that the product line of your chosen vendor will support such growth. Does the unit support multiple remote devices into a single host-support device? Can several multiplexed lines be combined into a single line or split to alternative destinations? Can users request connection to other users in a data switching mode? Multiplexers and concentrators do not mix well between vendors, so try to find a vendor with the features you need to grow.

• **Beware of buzzwords.** Most vendors advertise that they have a line protocol "similar to X.25 Level 2" or an "SDLC-/HDLC-like protocol." It is not even particularly useful to have a line protocol **exactly** like X.25 Level 2 unless it is also like X.25 at Level 3, since your host computer package is unlikely to support Level 2 only. To have something which is only similar to X.25 or SDLC is even less useful. X.25 is a good output protocol for concentrators because they will send it directly to a host computer. Multiplexers which use X.25 as an internal protocol can be used interchangeably as concentrators (by taking one of the pair away and running the high-speed line directly to the host) but may have a greater delay and a higher price than those that use a customized protocol designed for maximum efficiency in a multiplexer environment.

• **Direct bus attachment.** Some vendors provide special forms of multiplexers which at one end of the connection will attach directly to the channel or bus of a computer, replacing a communication controller. The DEC UNIBUS connection is the most common of these. Since the cost of a communication controller for the host is sometimes quite high in itself, combining a multiplexer with it provides additional savings. The risk is the possibility that the host will be replaced with a model that is not compatible with the multiplexer/controller.

• **Multidrop multiplexers.** Some multiplexers can be used on multidrop lines, offering the advantages these types of links provide to users of asynchronous terminals that normally cannot be multidropped. **Don't use multidrop multiplexers with multidrop terminals.** Delay with these devices can be considerable, even with little load. They work best with applications where a block of information is sent to and from the terminal rather than strings of single characters.

■ DIGITAL COMMUNICATION—SATELLITES, DDS & DIGITAL TERMINATION SYSTEMS

Analog systems based on the public telephone network have the advantage of being available between almost any two points in the country and throughout most of the major industrial countries of the world. But analog is limited by the characteristics of this same system to about 19.2K bps. Users who need higher data rates must look to digital transmission. Digital transmission is most often based on pulse-code modulation, or PCM, which is a high-speed sampling technique being phased in as the standard means of sending information on microwave trunks and thus through the "backbone" intercity/interstate phone network. Because PCM works equally well with digital data (it is itself digital-voice data that must be "digitized" to be carried) most of the analog system problems with noise, bandwidth, conditioning, etc are eliminated. Digital transmission, particularly satellite digital transmission, is becoming increasingly popular as the number of voice channels increases and the local system becomes more overloaded. **Figure 9** shows a chart of the relative costs of high-speed transmission facilities, demonstrating the importance to users of weaning away from the traditional analog circuits.

The basis of most digital transmission, whatever the service type, is the so-called **T1-carrier**. This is a 1.544-million-bit-per-second TDM channel that can be subdivided into 24 speech channels carrying PCM-coded voice at 64K bps or data at up to 56K bps.

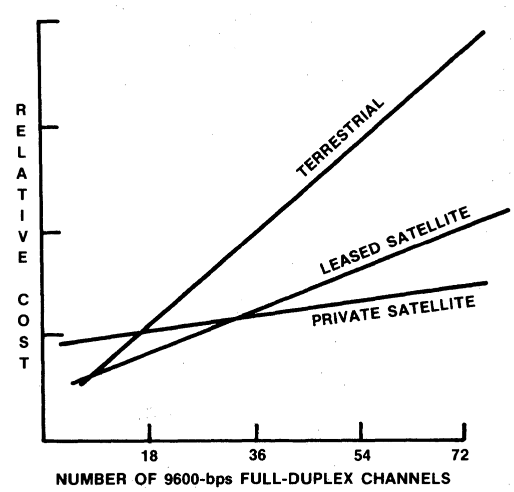


Figure 9 • satellite versus terrestrial costs.

Business Communication Networks—Putting It All Together

Each of the 24 channels can be divided down in any way that the users agree upon. For example, the 56K-bps channels can be time-division multiplexed to serve five users at 9600 bps.

Most users cannot directly utilize T1 data rates, although commercial TDM-type T1 multiplexers are available. AT&T Communication's ACCUNET Digital Service and SKYNET Satellite Services are available in most major cities at rates up to 3M bps. Digital circuits have no modems and an extremely low error rate, making them ideal for high-volume users. Digital service should be investigated by any users who operate in areas where it is available, if leased circuits are being considered, and modems have not yet been purchased. Be sure to check for competitive digital services as well—other companies offer a digital service, and satellite carriers may also be an option in some service areas.

Digital transmission is offered by satellite carriers through the use of a series of geo-synchronous satellites to provide high-speed communication potential thoughout the world, and it offers relatively low-cost user-to-user service between major population centers. Normally, a customer must connect to an access center (earth station) via a leased line. Satellite digital transmission is significantly less expensive than terrestrial for transcontinental usage and the cost trends are much more favorable, but satellites introduce a propagation delay because of the long distances involved (a hop is about 51,000 miles and takes about one-third of a second) which has a major effect on the performance of the channel.

Most high-performance protocols have error detection and correction capabilities, involving a periodic exchange of acknowledgements of correct reception. In this type of protocol, the sending station must eventually wait if these acknowledgements are delayed, since to continue to send would be to defeat the error recovery system and probably cause the other station to reset the link. This situation is sometimes called "closing the window." When a protocol is used with terrestrial links the delay in the communication path is very small and has less effect on the window than, for example, the modem characteristics. In satellite channels, a transmission and acknowledgement requires a round trip on a path with a third of a second delay, causing over a half-second of total delay. Protocols such as IBM's binary synchronous (BSC) which require that each block of data be positively acknowledged must therefore delay over a half-second for each block. This delay time may exceed the actual time used to send the block, so the effective capacity of the channel is dramatically reduced. Protocols such as SDLC/SNA and CCITT X.25 have a greater immunity to delay because they permit more data blocks to be outstanding without an acknowledgement. But even the 7-block window of these protocols is just enough at 9600 bps, and cannot provide full performance at 56K bps.

Users with protocols not efficient for satellite operation may employ a form of satellite delay compensation. These devices are actually protocol converters and transform the delay-sensitive user protocol to an internal protocol suitable for long path delays. This requires that the user protocol be emulated to the extent that the acknowledgements for data are delivered to the sending station by the compensator at the local earth station and **not** by the remote user. At present, several satellite carriers offer compensators as a part of their service. There are also a number of commercial devices that can serve as a form of delay compensator, but **most concentrators and multiplexers that claim to have a protocol immune to satellite delay will not render a user protocol immune also**. This means that the device will not introduce its **own** sensitivity but will not cure your own either. If you plan to buy a compensator, make sure that it really is one.

A new type of service is being licensed by the FCC to provide high-speed digital transmission. This service consists of a series of city-wide **Digital Termination Systems (DTSs)** which may be linked together by satellite paths to form a national Digital Electronic Message Service. Over 30 companies have filed with the FCC for authority to provide DTS, and the first group of licenses have been issued. The DTS concept, shown in **Figure 10**, uses microwave distribution locally to bypass the leased phone lines used for conventional and satellite carriers. The elimination of the **last mile** local loop that reportedly causes most of the problems in data transmission is expected to improve performance and cost, and to shorten wait times in congested metropolitan areas.

Current digital services are essentially forms of leased-line service, and thus, most appropriate to users with relatively high data volumes. Some satellite carriers, requiring user-purchased earth stations, are usable only by the very largest firms. Pricing on

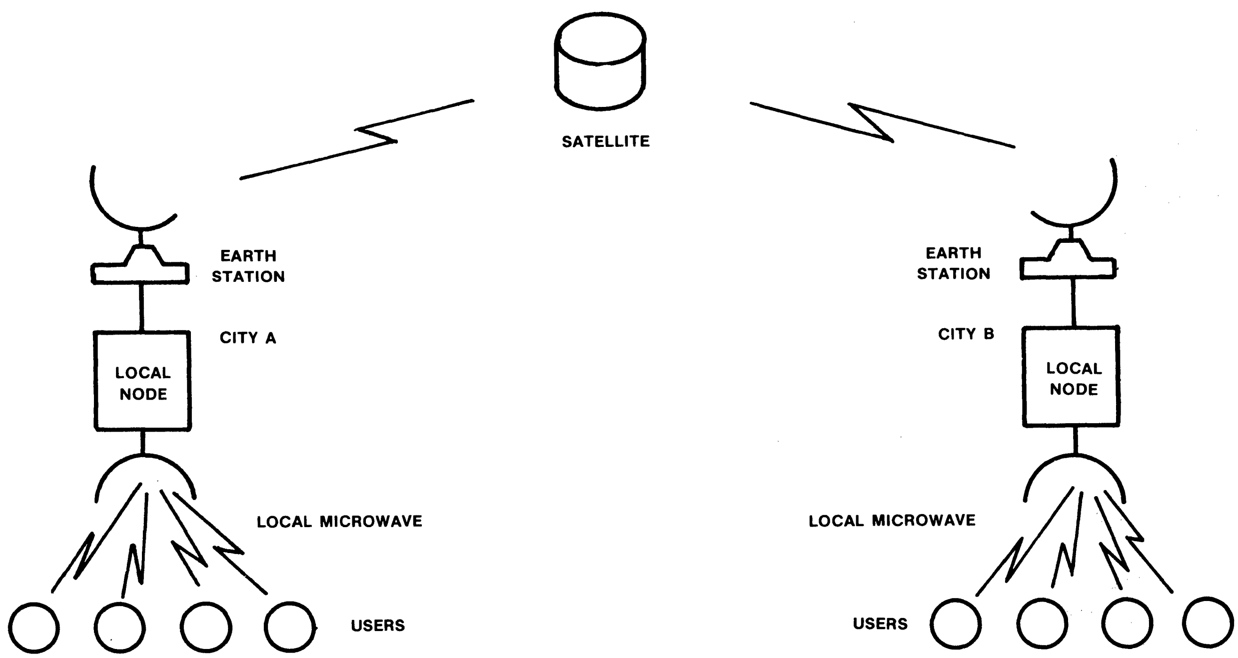


Figure 10 • a typical digital termination system.

Business Communication Networks—Putting It All Together

DTS is not yet available, but it appears that it will be competitive with leased lines.

■ PUBLIC PACKET CONNECTIONS

Public packet networks are to dial-up service what digital services are to leased—and more. Phone service is billed based on time and distance, making dial-up links expensive for users who call frequently or who must call long distances. Public packet networks are generally billed on a combination of time (for dial-in users rather than those who lease a local connection to the network) and data volume. Because the charges are distance independent, some organizations that must collect information nationally may find that public packet networks offer them the only viable method of data communication.

Public packet networks are made up of a heavily interconnected series of computer data switches called **packet nodes** or **packet engines**. These switches collect user data in bundles called **packets** and transfer them from node to node until they eventually reach the node attached to the destination. Terminals may access such networks by dialing into a public access port or by leasing a **local access line** to the network. Host computers, since they normally must be online to the network at all times should a terminal "call in," are connected by leased lines.

Figure 11 shows a chart of the relative line economics of leased connections or dial-up circuits versus public packet. As you can see, nearly any connection more than two states in length can be expected to be less expensive with packet networks. Why use leased lines then? First, very high-volume users who can expect to fully utilize a high-capacity leased trunk will probably find it less expensive. Packet charges are based on allocating that trunk cost over many users, and of course a reasonable profit is added. Other problems that can impact potential users are more subtle: packetizing problems and network delay.

Packet networks switch packets, so if your device doesn't happen to communicate in blocks of data (asynchronous terminals, for example, normally work a character at a time), either you or the network must form the data into packets. This can be more difficult than it sounds, because grouping data into messages introduces a delay while the data is being collected into the packet. If the computer-to-terminal dialog happens to require that one of the collected characters signal an action, and that character is being held up for packetizing, no action will take place and the operator will assume a system failure. For example, if you are keying an account number of up to 10 digits followed by a carriage return, the computer will probably look for the carriage return as the signal to process. If the packet network does not recognize that the return is the last character you will key, and holds the data so that a nice large packet can be collected, the operator will never receive a response. The process of building packets is called "packet assembly/disassembly" or PAD; it can be tuned to a particular application by varying the

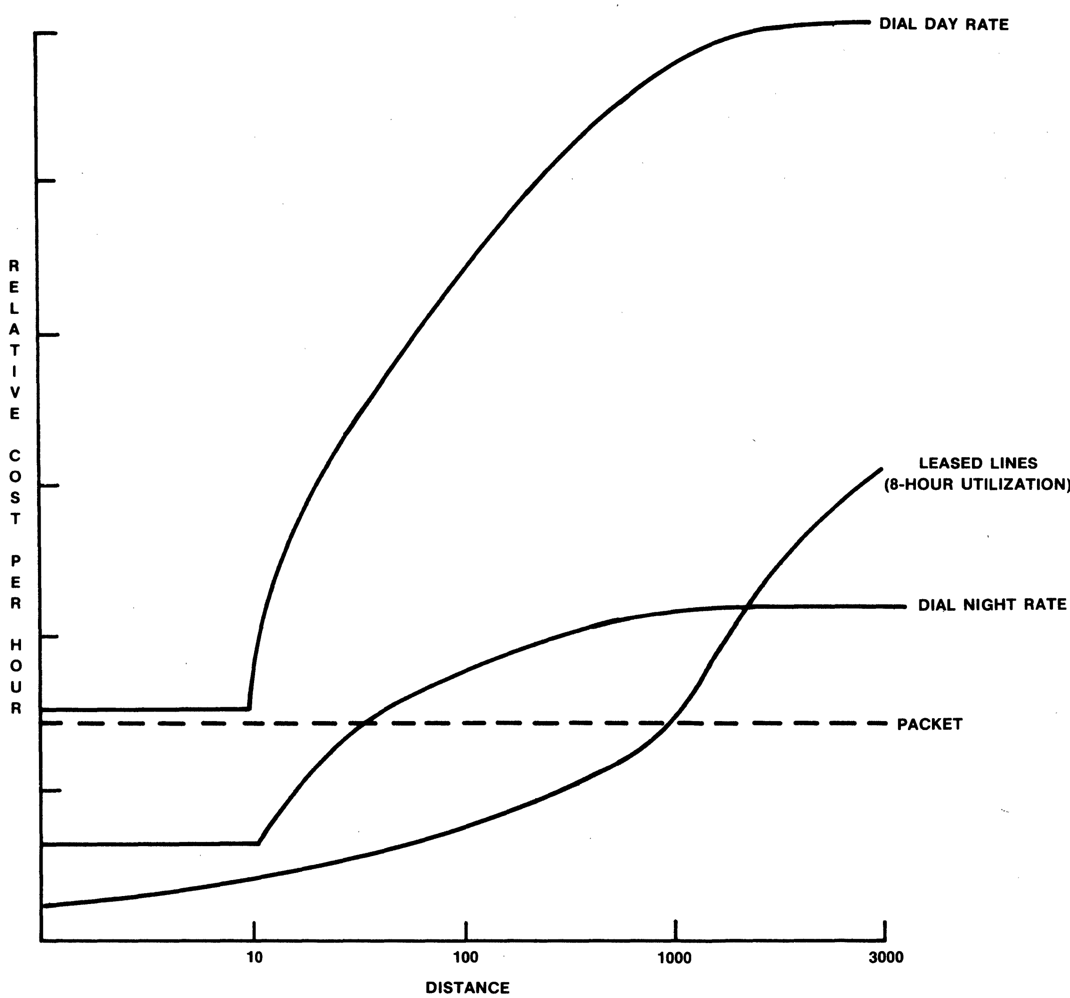


Figure 11 • relative communications costs.

Business Communication Networks—Putting It All Together

value of **PAD parameters**. Pad processing is defined by a series of international standards, X.3, X.28, and X.29, but various network operators such as GTE Telenet may also have their own.

Host computers can attach to a packet network directly in packet mode, using the international packet interface standard known as X.25. Attachment in this way normally reduces the network charges because X.25 can support multiple conversations over a single link. If your computer or front-end processor is **certified** by the network operator, it can be attached directly to the network, otherwise it must be connected through a **network engine** which adds significantly to the costs.

Network delay may be a factor in the use of packet networks. The time it takes to collect the data into packets and to handle the packets in each node can be considerable, creating a problem not unlike the satellite delay problem, but complicated by the fact that the shared trunk facilities of the network may not be able to allocate enough resources to your call to support the rate of data delivery your host or terminal could supply. Users with high data delivery requirements may find that leased line service is necessary, and it may even be less costly.

A newer form of public packet service is available, offered by the unregulated subsidiary of AT&T, Information Systems. This service, called Net 1000 Service, is designed to provide public network features to users who could not normally afford the public packet charges, a feature the company calls **low entry threshold**. Unlike most public networks, Net 1000 Service has a message storage mode that lets users deliver batches of data to the network at one point in space and time, and to extract them at another. If the application programs and terminal operators can adjust to the fact that no actual terminal-to-system dialog is taking place, this could allow a series of dial-up terminals to enter data to be accessed later by a dial-up host.

Public networks are not everyone's answer. The following questions can help you determine their applicability to your network needs:

• Is your network geographically distributed? If you have local connections, public packet is probably not your best choice.

• Is your network currently supported by dial-up lines, or do you expect the use of a new network to involve mostly short connections? High volumes and long connect times are less economical in public packet than low volumes and short calls.

• Does your computer already support X.25? Can it be made to do so? Generally, the most economical means of attaching a host to a network is through a certified X.25 interface. If you cannot do that, the alternatives may be too expensive.

• Is packet service available throughout your network coverage? Telenet, Tymnet, and Uninet each cover nearly 300 major service cities, but you may require connections elsewhere.

Mixing packet and dial connections is possible, but not always easy with all host computers. Check your system's capabilities before you commit to anything.

■ NETWORK MANAGEMENT & CONTROL

Most users depend on statistics from and features of their host computers or front-end processors to control their networks. For simple network structures or for users who can adopt a single-vendor or plug-compatible network architecture this may be the best solution. For users who have begun to introduce such devices as multiplexers or concentrators into their network, it cannot be fully effective because these devices are unknown to the computer, and thus, uncontrollable. Where transparent devices offer network control features, these features may offer an alternative to host control, but it is difficult to integrate several different control packages satisfactorily. In general, if you cannot get the desired level of network control or statistics from your host access method or network nodes alone, you should consider a separate and comprehensive network control system.

Using a network control begins with defining precise objectives. As a user, you have various network control and management features available, some of which may not be applicable or cost justified. Some of these options are:

• **Switch or plug.** This capability, called **patch panel**, provides connections between lines or modems and computers or terminals. This allows you to switch lines or ports on the system when a failure occurs, or to change the division of resources between systems.

• **Line statistics.** This capability provides a way of gathering traffic information and error statistics on a link. The statistics are useful in communication planning, in allocating costs, or in detecting a rising number of line errors which could indicate a deteriorating link.

• **Line monitoring.** This consists of a data line monitor that displays data and control activity on a line, and optionally "T" connections to allow each line to be monitored without disconnecting it. Line monitoring is useful only if the users of the facility are relatively sophisticated.

• **Network monitoring.** This allows the relatively unskilled user to determine the status of lines and other network elements from a central point. The information is normally relayed in more user-friendly displays than line monitoring would provide, and the status of the entire network can be determined quickly. This capability is normally available only through the use of a computer-controlled device, either a network node itself or a separate system with links into various network devices. Most systems such as AT&T-IS Dataphone II Service use special diagnostic/control modems that have a separate narrowband signaling channel (**secondary channel**) for the exchange of network status information.

• **Fall-back and recovery.** Advanced network control allows recovery from failures to be initiated through global, predefined steps under user control or automatically upon failure detection. The recovery may be limited to failures in the links, or may extend to more complex rerouting associated with node or host failures.

Most small users will not require complex network management. If you have less than 10 lines into your computer and do not have a requirement for high reliability and extensive spares, you probably don't need any network management capability beyond the limited statistical information on the lines which probably is available from your host.

The initial step in network control may be either to a patch panel system to permit line switching or to a means of line testing and monitoring, such as a data line monitor or a line statistics monitor. These can serve as a way to gather information on the condition of the line and to adjust the communication configuration when a problem is detected. One of the primary benefits of this step is associated with use of leased lines. If precise information on the error rate or type of failure experienced is available, and can be demonstrated to a repairman on request, phone company cooperation in correcting a problem with a leased circuit can be improved dramatically, increasing your uptime.

More sophisticated network control is likely to be expensive, so the best alternatives should be explored. First, does your access method provide any form of control? Network architectures such as IBM SNA will offer considerable flexibility in routing and fall-back. Second, some private network devices such as concentrators may provide network control. While this type of control may be difficult to integrate into the existing network if host control is also supported, it may be cost-effective.

A final option is the acquisition of a full **network control system** such as is offered by AT&T-IS, Codex, General DataComm, Timeplex, Kinex, Paradyne, or Racal-Milgo. These systems operate in conjunction with vendor-supplied modems to provide a means of gathering network information and conducting testing from a central location. The control information is carried on a low-speed secondary channel which may either share data path transparently or use other pins of the RS-232C interface.

If you are evaluating network control and network management systems, be sure to consider the modem implications if you have a network in place. Some systems will not operate with other vendor devices, and users rarely install modems with network control capability unless they install the entire system. This means that you probably do not have the modems already in place. Network control in a modem is almost certain to increase its cost over comparable units without it.

The following checklist is useful when evaluating network control systems:

Business Communication Networks—Putting It All Together

• **How many individual network elements will the system support?** Don't buy something you will outgrow.

• **What type and speed of modems are supported?** Your primary data communication circuits all have speed and other modem characteristics associated with them. Be sure that the system you choose will support **all** your devices. For example, many of the systems will operate only with 2400-bps to 9600-bps synchronous modems. Chances are that networks with low-speed circuits could not justify the costs of such a system, but there are obviously exceptions.

• **What type of information is available?** You should be able to get alarm reports on problems, line statistics (character and message counts), and more technical displays such as the "star" pattern used to detect line or modem problems with phase jitter.

• **What is the output device?** Reports on a printer or a printer port on the system is a must for records of activity.

• **What types of lines are supported?** Will the system work with multipoint circuits?

• **Is remote testing supported, and in what form?** You should be able to set the remote modems into loopback for testing.

• **What is the total cost?** Be sure to get a quote on the equipment, the modems, any special cables, and the monitoring device. Include the costs of any special bridging needed to route the special control paths around existing network devices, switches, etc.

■ TRENDS IN COMUNICATION SYSTEMS—PLANNING FOR THE FUTURE

The explosive rate of growth in data communication technology demands that users plan their present networks in the light of projected trends, both in their own usage and in the field as a whole. The first step is to project your own demands on the communication network and the data processing facilities it supports, and the second to evaluate the trends in communication that might impact your present and future.

Standards in data communication are probably **the most significant force users should consider**. Organizations like the International Organization for Standardization (ISO), the Consultative Committee for International Telegraphy and Telephony (CCITT), the Institute for Electrical and Electronic Engineers (IEEE), and other have been working to provide definition and direction for the rapid growth in the industry. Standards are being worked on for network architectures, local area networks, integrated digital networks, message switching, videotext, and other areas. If you are moving into an area where standards exist, try to be sure your equipment complies. If standards do not yet exist, try to find out what progress is being made.

A second area to guard is that of technology. Microcomputers and integrated circuits are finding their way into nearly every aspect of data communication. Certain products, such as modems, have actually tended to get less expensive through time due in part to these advances. Before you buy in anticipation of a future need, check the cost trend for the product type and hold off if costs are declining. In buying for a current need, try to select a product which has a state-of-the-art design to lessen the risk of technical obsolescence.

Finally, evaluate the trends in the environment; interest rates, line costs, satellite charges, etc. Some of these trends are shown in **Figure 12**. In general, the costs of terrestrial links are almost certain to increase significantly because of the competition for microwave and cable space. Until fiber-optic cables become common, satellite carriers will be more and more economical in proportion to their earthbound counterparts.

For the far future, most experts believe that network architectures such as SNA and DECnet and international standards such as CCITT X.25 or the OSI Reference Model for Open System Interconnection will be more important in product development and thus to users. Labor costs, continuing to rise, will increase the need to boost productivity with point-of-transaction computing power, so distributed processing will increase. So, obviously, will data communication. The value of information will increase, making it important to collect and disseminate it efficiently and to make studied decisions in areas where the quality of information might be affected, such as its timeliness.

■ SUMMING IT UP—DEFENSIVE PLANNING

Probably the central requirement for future planning in the current environment is flexibility. If you select a path, make it a decision which provides you with as much latitude as possible and which tracks what seems to be the center of the fan of possibilities that grows in width as it extends into the future. Being exactly right, exactly optimum, is unlikely in a field as rapidly changing as data communication, but being excessively wrong is relatively easy.

New users of data communication should plan their data communication facilities around their total data processing strategy, and set a long-term goal in terms of a network architecture or access method that can be worked toward with reasonable costs in the present. Try to adopt the most standardized solutions you find, not the most elegant.

Current users should evaluate their network changes in terms of a long-term strategy for migration and not just on the basis of lowest cost. Where there seems to be a clear direction, follow it. Otherwise, avoid committing yourself even if it means living with existing equipment a little longer.

Finally, all users must conduct realistic postmortem evaluations of failures, not to access blame but to prevent their recurrence. Each time you make a mistake, examine the decision process that led to it, the implementation process that brought it about, and the

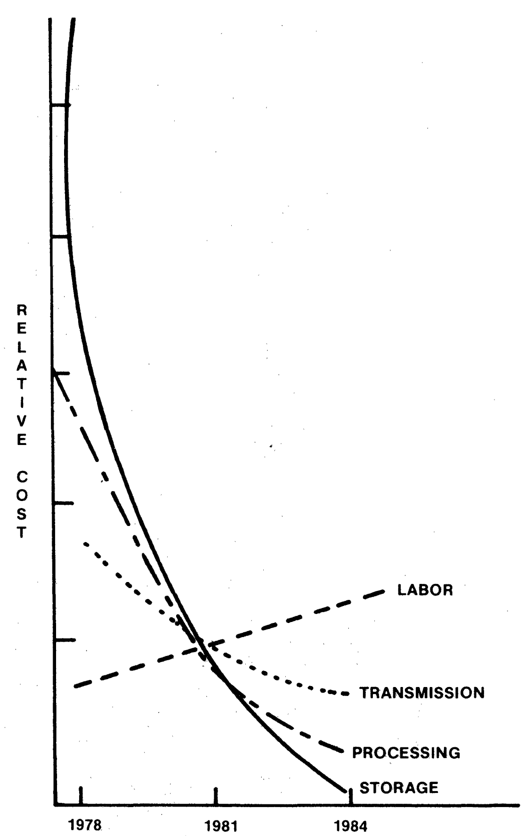


Figure 12 • cost trends in information processing.

Business Communication Networks—Putting It All Together

operational environment in which it failed.

Data communication is a complex field, and integration of the network or of network to computer system is a task to be approached with care. Only by being as knowledgable as possible on your own needs, the technology, the trends, and the products can you hope to make decisions that will serve your company today and in the future.

• END

Microcomputer Buyers Guide
Personal/Microcomputers Used In A Corporate Environment

The gains in productivity which could result from the use of a microcomputer are sending businesses scurrying to obtain these systems in record numbers. Many corporations have found that microcomputers can make them more responsive to current business needs, eliminating the often long delays associated with developing applications on large mainframes. But the personal computer may bypass more than programming development delays. Without control, companies may acquire multiple incompatible units and spend hundreds of thousands of dollars on redundant software. Worse, the spread of the systems may disperse valuable management information in a way which makes meaningful corporate MIS impossible to develop. While these risks should not deter the application of the microcomputer to a corporate environment, they do mandate prudence in the evaluation, selection, installation, and control of these small but very valuable business resources.

This report will explore the organizational and administrative issues associated with the employment of microcomputers in a corporation, then cover the technical characteristics of these systems as they relate to their application in large businesses.

Data processing managers may well find it helpful to pass this report along to user departments considering their own local systems. Additional copies of this report are available upon request to Marketing Director, Data Decisions, 20 Brace Road, Cherry Hill, NJ 08034. Telephone 609-429-7100.

THE CHALLENGE OF MICROCOMPUTERS

When personal-sized computers became popular, many could see that their low cost and high performance would revolutionize the application of computers to small businesses and even to the home environment. The growth of microcomputer shipments worldwide is expected to top the 11 million unit mark by 1991, up from about 2 million in 1982. The majority of the systems sold over $1,000 are not going home or to small businesses, but rather to large corporations as personal workstations.

As computer systems become more powerful and sophisticated the quality of programming resources needed to develop applications for them rises. Today most large users gladly purchase packaged software to preserve their scarce professional programming resources for large and vital projects. The application of mainframe technology to large projects has left behind many of the small tasks which can smooth the operation of a business or improve the quality of its performance. Until recently the only alternatives to a large system were remote computing services, but these often proved unsatisfactory due to lack of local ability to control or modify the systems and functions. But the microcomputer has changed all of that, and in doing so has placed business in a delicate position. Does your company permit individual product managers to purchase raw materials independently? Probably not. The cost to the company of separate purchases in terms of volume discounts lost, would in itself, justify central purchasing, and additional problems with compatibility and standardization would undoubtedly occur.

Patterns of change in the labor force due to a combination of technology and economics have begun moving workers from primary production tasks where conventional computerization can affect their productivity to "knowledge work" where free-form access to information is the key to productivity. Potential corporate savings to be realized by increasing the productivity of these new types of workers is estimated in the billions of dollars. The key to realizing these savings appears to be the utilization of an INFORMATION STATION rather than a conventional WORK STATION; a point through which the knowledge worker can gain access to masses of data for correlation and manipulation—a window into the information bank and processing power of the corporation. This information station should also have the processing power and programming support for local digestion of data without a requirement for extensive programming so that the efforts of the worker are not diverted from the solution itself. The microcomputer is already more than a match for most of these requirements, and it is getting closer to the ideal with every new industry development.

When a small business or an independent professional selects a microcomputer the selection issues are well defined and local to the selector. Suitability for the specific application is the primary issue, and certain technical risks can be taken in the purchase if they can be clearly defined and measured against the economic or other gains they may provide. In a large corporation it becomes more difficult to measure the trade-off because the purchase of a single system is just a part of a larger pattern of evaluation and purchase, the net effect of which will likely be the acquisition of hundreds of thousands of dollars worth of microcomputers. If each departmental user in a large corporation judges local issues only, the cumulative odds that the equipment will be unmaintainable or incompatible are excessive. To the individual user, an ideal system would be one custom-selected for the parameters of the task at hand. To a corporation, the ideal system is one which will perform in a cost-effective manner across ALL the tasks to which it might be assigned by a number of departments. Left to their own devices, departmental users in a large corporation might obtain dozens of different incompatible products for their individual needs. Following some centralized corporate guidelines—albeit guidelines that acknowledge the reality that several different (possibly incompatible) systems are necessary in order to support a host of applications needs—the same corporation might turn to 2 or 3 different models that greatly simplify and order the overall environment.

In this report we will deal constantly with the dual perspective suggested here: the buyer as representative of a problem requiring a timely solution and the corporation as an arbitrator of technical and business standards which must be adhered to in order to protect both the investment and the integrity of the whole. The question of corporate use of microcomputers will be addressed first from a policy point of view, then from the perspective of evaluating the characteristics of such systems as they might affect the individual users.

Microcomputer Buyers Guide

Personal/Microcomputers Used In A Corporate Environment

MICROCOMPUTERS—THE ORGANIZATIONAL BUYER

Microcomputer use within a corporation is usually started by a single professional or manager who finds an application which seems to justify the system. Because these applications tend to arise at the first-line management level where decisions are likely to be made more quickly and action is expected in a short time frame, lack of predefined corporate policies may produce improperly researched solutions. A local project may be formed with a line manager as the reviewing agent and a purchase made within days. These applications tend to fall into several categories.

• Local processing needs such as work flow scheduling, departmental budget control, special financial projects for clients or management, or preparation of proposals or other formal documents. Such needs are often addressed first in terms of word processing, then through exploration of business computer packages for a word processor, and finally as justifications for a microcomputer system.

• Professional productivity aids such as recording of meeting schedules and travel calendars, expense accounting, lead tracking, client information maintenance, and individual sales tracking. Most of these applications are suggested either by seeing a similar application elsewhere or by the professional's own pondering of the problem. Viewing the need as personalized rather than company-wide, such a prospective user may actually purchase a system privately to do the job.

• Ad hoc, short turnaround projects which are motivated either by business opportunity or government regulatory requirements. If these needs are local in scope the responsible management may not desire to lose control of their satisfaction, and a timely response from the corporate data center may not be possible because the job is too small. Many businesses underestimate at the corporate level the importance of being responsive in such situations, leaving the local management to cope with the problems with their own resources, perhaps a small computer.

• Management and executive programs to improve computer literacy and the comprehension of the application of computers to the business. This may range from actual training in computer applications based on personal systems or even programming training to a cosmetic purchase of a system to adorn the desk and project an image of a high-tech company to all visiting clients. Since these applications are very unfocused they are probably the most difficult to control and the most dangerous in terms of accidently setting precedents.

Purchasing department or data processing department projects may also be initiated to anticipate computer use and provide advance guidance through evaluation of alternatives and recommendation of specific vendors/products. This effort is not normally due to insight on the part of a purchasing manager or data processing professional; but generally arises from a user department inquiry. Properly supported from the outset, this type of pre-emptive study can lay a firm foundation for the use of microcomputers within a corporation, reducing costs, avoiding pitfalls, and improving control.

To a line manager with enough budget authority to purchase a computer, a private resource may be an opportunity to escape what appears to be the callous indifference of the corporate data center in developing and running the department's applications. The threat of "going it alone" is calculated to intimidate the data center and insure better treatment; but at the same time there may be a powerful need to gather all the elements of the department's business activities into the department's own organization—to gain the control which the department feels should accompany its responsibilities. Uncontrolled growth of local processing can cause problems in a corporation even when the motives are the best. An adversary relationship with the corporate data center is hardly the best motive, even if the interests of the prospective departmental buyer are clearly not being served at all. The alternative to this is the definition of a corporate policy on small computers.

MICROCOMPUTERS—THE CORPORATION

Corporations normally have central corporate data centers to serve the processing needs of the operating departments. Centralization has grown partly from the "economy of scale" which says that a large computer to serve multiple applications will be less expensive than several smaller ones, and the second has been the marketplace reality until recent years which was that adequate personal/microcomputers simply did not exist. Another reason for centralization has been the need for a specialized data center environment—climate control, conditioned power, and security of confidential information. The need for highly specialized professionals to control the development of applications and establish the basic computer environment in which work will be done has also favored a central data processing organization. This group must set standards for development, plan the technical growth of the facility, and determine the costs and benefits of applying the computer facilities to the various business tasks. The goal of the standards and controls applied by this central agency is the reduction of the corporate data processing costs and the assurance of a consistent development standard so that the information gathered for the operation of one organization can be processed and correlated with similar information elsewhere to serve as a basis for corporate decisions.

Microcomputers in a corporate environment pose a dual potential problem. The acquisition of local computer power by individual departments may bypass the application review process and result in unjustified expenditures and poor design and implementation. It may also fragment the information collected during the operation of the business in such a way as to make it unobtainable from a central point, starving the management information system development process. When minicomputers became widely available, many companies purchased a separate computer for each profit center or geographic location. One major bank began the critical 1980's swings of inflation and recession with 11 corporate data centers, 22 separate computers, and 9 different computer vendors. Bank clients which had an integrated management system could determine their exact positions with the bank in any area at any time, but the bank could only assess their positions on a class-of-business basis, and then only as of closing the prior day. Why? Because the commerical loan, trust, commercial DDA, and other departmental systems were on separate and largely incompatible computers which were never designed for the integration of all information. Most minicomputer users had long since learned from this uncontrolled dispersion, but microcomputer users today are non-computer-professionals who never went through this learning process before. The potential thus exists to repeat the errors of the past, with consequences at least as disasterous.

Many corporate data processing executives will remember clearly the problems and issues surrounding the central/decentralized debate which was touched off by minicomputer availability:

• Loss of central standards control. Most corporations realize that the long-term maintainability of applications depends on a central support organization and in turn upon adherence to a consistent set of standards. There are also standards for the evaluation of the suitability of an application for automation, for adherence to corporate return on investment objectives, and for technical issues such as RF interference, compliance with electrical codes, etc. When a local department purchases equipment, without the benefit of professional advice, they may do so without knowledge of these standards. One user found that a mysterious source of interference to important communication facilities was caused by several personal computers owned by executives and brought to the office to run an expense account reconciliation program. The units involved were old models which did not meet current FCC RF specifications.

• Redundant purchases. Often 2 or more departments will purchase computers when a single unit would serve all the users, but the problem is more severe with software and peripherals. While software restrictions vary, it is often possible to use a single copy of a program on multiple systems through payment of a special, low-cost, multiple-copy license fee. Without knowledge of one another, many users may purchase the same software at full price.

• Incompatibility. The major technical problem with decentralized purchase of equipment is incompatibility. There are

Microcomputer Buyers Guide
Personal/Microcomputers Used In A Corporate Environment

many types of microcomputers, most of which will have some restrictions on the exchange of information with other systems. Incompatibility may be in the form of MEDIA, meaning that information stored on one may not be read by another, in the form of STRUCTURE, meaning that the format of the information is not compatible although the storage media could be read, or in the form of PROGRAM, meaning that the programs for 1 system will not run on the other. Connecting computers together with communication lines may eliminate media compatibility issues, but just transferring a data file or program will not necessarily make it usable.

• Loss of central purchase control and volume purchase power. Most user organizations have little experience in purchasing a technical product. Issues such as service and warranty may not be properly covered, resulting in a loss of service and additional expense. In addition, the system as ordered may not be satisfactory for the application—not all computer systems are shipped with a terminal for operator control, a printer, or even programs to run. Individual purchases may also lose the volume discounts often available to large users, sometimes 25% or even more. Finally, individual purchasers have less leverage with a vendor. If problems arise with a system, a large user with a record of significant purchases can often get a quicker solution than a single department who purchased a system from a dealer not used by any other part of the company.

• Improper operating environment. Even small computer systems require reasonably smooth AC power connections and temperatures in the normal range for human comfort. They generate heat, sometimes in considerable amounts, which can turn a small and poorly ventilated office into a sauna. A particular problem is their sensitivity to static electricity, which can cause problems ranging from an annoying periodic restart of the system and loss of data to a static-induced hardware failure causing hours or days of down-time and costing hundreds or even thousands of dollars to repair.

• Loss of security. Confidential information stored in small systems is very susceptible to piracy because the systems are rarely in an area where physical security could prevent unauthorized access. Many systems are open workstations and accessible to casual visitors. Modular office planning may create inviting "hidden" areas in which workstations can be used without observation. Furthermore, the electronic storage media itself, floppy disks for example, are easily hidden and transported. It may be difficult to remove a listing of customer information several inches thick from the marketing department, but a single diskette half the size of a sheet of paper might easily fit into a coat pocket.

• Dispersal of information. In an ideal computer environment, all the information which is collected in the normal business processes is stored in a way which allows it to be correlated for management review, even to the extent of searching for new relationships and structures which might be the key to a profit opportunity or provide insight as to how to reduce expenses. Microcomputers can cause the information to be dispersed among a large number of systems where its existence may not be known and where incompatibilities make its analysis at a central point nearly impossible. Can you do an expense account study on the potential savings of using private automobiles versus public transportation to certain cities if all the travelers have their records on their own computers? Dispersal of information can also hide significant problems from management, possibly until it is too late.

COMPATIBILITY is a basic issue of many of the problems outlined above. In evaluating computer systems, a corporate user must consider 3 levels of compatibility as discussed below:

• **Compatibility with other small computer systems.** Local processing needs are often considered unique, but many departments find that the basic problems of work scheduling, activity tracking, expense management, and financial projection can be solved by a common system for all users. Using the same, or a compatible, computer will allow the users to exchange data and programs with minimum effort. It will also permit the stocking of accessories such as peripheral devices, storage media, and supplies in a common pool, and the interchange of these articles among users. It is important to note that typical expenditures for after-sale items such as supplies, extra software, maintenance, etc are about the same amount as was spent to purchase the microcomputer; with about half that figure spent on media/supplies above. The ultimate in compatibility is having the same brand of system utilized in all locations. When different models or different vendors are used it is necessary to determine the degree of compatibility through research and possibly testing.

• **Compatibility with the company mainframes.** The central data center should always be considered the primary source of computer power in a corporation. Ideally, each small, local computer should have the ability to interact with the mainframe in some manner for the exchange of information. This interaction may be via the direct exchange of storage media such as disks or tape, or may rely on communication links. In some cases, a small computer can also serve as the terminal on a larger system. There may be issues of program compatibility between the systems. Compatibility with large mainframes requires accommodation on both sides; the corporate system may have to be upgraded in some manner to support the connection to or interchange of media with ANY microcomputer. The cost of this enhancement may vary significantly depending on the type of microcomputer purchased.

• **Compatibility with external databases and sources of information or processing power.** There are many valuable sources of information available on a subscription basis, from stock market information to legal precedences. There are also service bureaus which provide for remote access to the raw power of very large computers. The ability to connect via communication line to these resources can enhance the value of a microcomputer system, but such interconnectability should not be assumed; some systems have no practical communication capabilities. An examination of the corporate issues shows that they divide into 3 distinct groups, balancing against the pragmatic concerns of the individual departmental buyer:

• Administration—Getting the best price, preserving an orderly relationship with vendors, securing favorable delivery, service, warranty, and contract terms.

• Standardization—Establishing requirements for the technical features of microcomputers, reviewing and recommending adequate vendors/products, developing application guidelines, formalizing documentation requirements for microcomputer applications.

• Policy making—Establishing a corporate position on the use of microcomputers, the relationship with the corporate data center, the degree of autonomy which local departments may enjoy in purchase decisions, security provisions that must be adhered to, and the degree to which valuable business data will be permitted to exist apart from and isolated from the main corporate database.

Potential microcomputer applications in a large company seem to develop quickly and require a fast decision. This short decision cycle is often used to excuse proceeding with a single purchase on the grounds that there will be time later to review overall policy for other purchases. Rarely is the time ever found. Once equipment purchases begin the chances of ever having a fully controlled environment are reduced and precedences for "local control" are set. If any form of company policy is to be set, it should be set IN ANTICIPATION OF USE and not after use of microcomputers has begun. The policy must be set by a combination of user effort and technical and administrative staff participation. It will require an evaluation of the potentials for use of microcomputers within a company and of the features and feature trends of the systems themselves.

DEFINING SYSTEM REQUIREMENTS IN A CORPORATE ENVIRONMENT

A staff study of each individual clerical or professional task in a company to judge the benefits which might be gained by the application of microcomputers is clearly an impossible task, and even if such a study were to be made it would be obsolete well within a year. It is more important to study the STYLE of the business; its relationship with its market, its way of using information, and its existing technical resources. These global factors will define the framework for a business plan in applying microcomputers:

Microcomputer Buyers Guide
Personal/Microcomputers Used In A Corporate Environment

• Integration of activity. Figure 1 shows 2 dramatically different business structures found in corporations. Some businesses operate in a very "compartmentalized" manner, with organizations interacting based on formal rules and at specific points. Others are by nature more unstructured, with many exceptions to procedures and frequent need for ad hoc "task forces" made up of representatives of many departments. Where an organization is structured and its interaction is predictable it can be given more latitude in its approach to its tasks because it can be treated as a "black box" which produces a specific response for a specific input stimulus by the rest of the corporation. Free-form organizations are most often companies which are new, high-technology, or in very service-oriented fields. They must operate more often across formal organizational lines, and anything which is done in one area even for internal use will probably affect other areas. Free-form organizations are the most likely to be damaged by the misuse of microcomputers because the impact of problems such as incompatibility are difficult to forsee but potentially devastating.

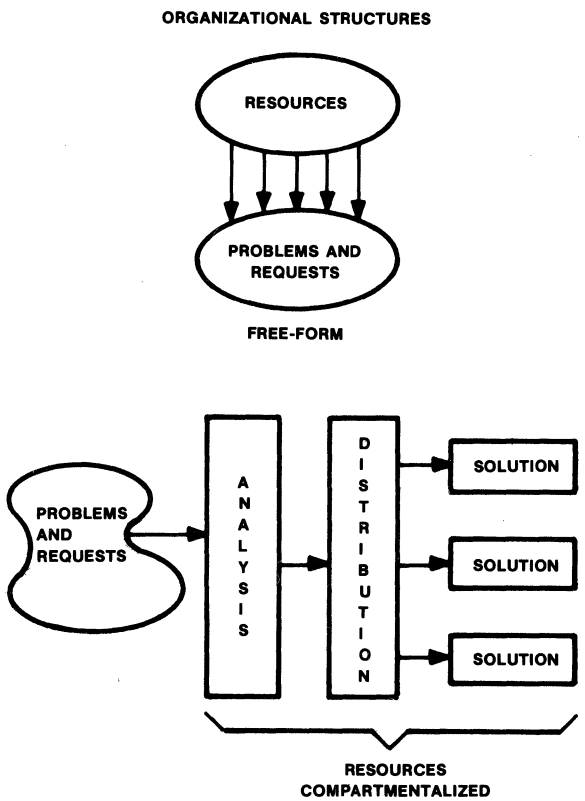


Figure 1

• Information usage. All organizations collect information which is TRANSACTIONAL in nature; sales orders, purchase requests, invoices, payroll changes, inventory transfers, etc. If this collected data is used along SYSTEM lines, keeping payroll data associated only with personnel/payroll reports and activities for example, it is more easily isolated in a small computer without impact on the business as a whole. Recent trends in the development of management information systems is leading users to become more INFORMATIONAL in nature; correlating such information as the rate of overtime pay against the rate of shipments or the absenteeism of first and second line management with the rate of system inquiries. Such firms may find that the isolation of key data in a microcomputer will make these SPECULATIVE CORRELATIONS difficult, or worse cause them to provide incomplete and erroneous results.

• Competition. If your business is highly competitive you must constantly monitor profit margins and expenses and must be responsive to problems or opportunities. Moving computer power out to the key workers may help such an organization react faster and plan better in the day-to-day crises, but may also result in a complete loss of control. Such users demand a tight integration of their microcomputers and their data center equipment.

• Current data processing strategies. The corporate data center is normally structured as shown in Figure 2, either for the INTERACTIVE service of many terminals within the business for online entry and inquiry or as a BATCH PROCESSING point where data collected elsewhere is processed in bulk. Interactive systems are more easily integrated with external computer devices because they already provide a form of workstation service to the users, making technical issues of attachment to the host computer easier and reducing the operator learning curves. Where a business has a pure batch environment microcomputers can threaten the modernization to an online system by reducing the primary motive for such a move—improved customer service through interactive maintenance of records. It is also more difficult to transport data from a microcomputer into a batch system because there is little media compatibility between small systems and mainframes. In general, BATCH USERS SHOULD EVALUATE THE MIGRATION TO AN ONLINE SYSTEM BEFORE CONSIDERING A MAJOR COMMITMENT TO MICROCOMPUTERS.

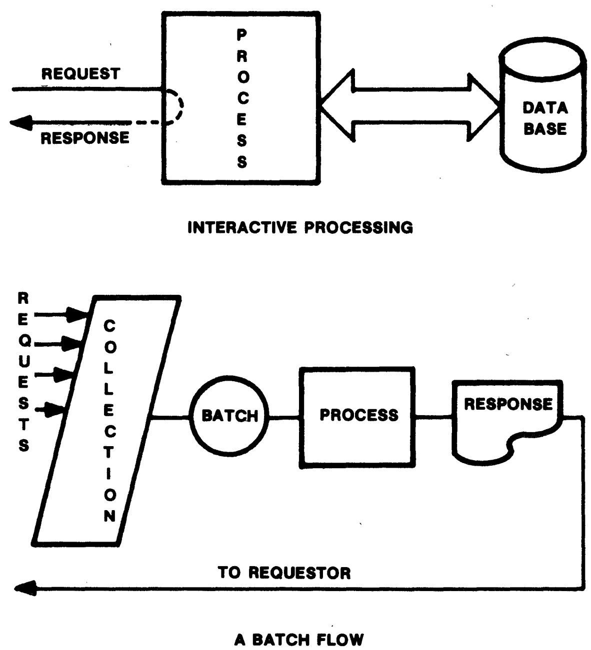


Figure 2

• Organizational structure. A company with many workers of equal status and similar job functions may find that the purchase of a number of microcomputers will be more expensive then the purchase of a single minicomputer and 1 terminal for each would-be microcomputer. This is particularly true when all of the workers do much the same job: order taking, brokerage, proposal development, etc. In most cases the type of information collected in this environment will require correlation into a single data bank in any case, making distribution of information into separate microcomputer systems impractical.

Microcomputer Buyers Guide
Personal/Microcomputers Used In A Corporate Environment

• Regulatory issues. If your business collects information on activities which are subject to any form of government regulation you have already come to accept the requirements for reporting. A relatively simple request can become a major project when the information needed is scattered across a dozen microcomputer databases. A firm rule: DON'T LET INFORMATION ON REGULATED TASKS OR BUSINESS ASPECTS GET DIVIDED. If you must use microcomputers for collection or use of this information, MAKE THE FORWARDING OF THE DATA TO A CENTRAL POINT A KEY ASPECT OF THE PROJECT.

If you examine your organization in each of the above areas, the results will help define a basic strategy for the employment of microcomputers. The possible strategies and the considerations which support them are as follows:

• Independent microcomputer strategy. Businesses may employ a strategy of purchasing independent microcomputers having relatively little integration with their data center or with other systems if they are COMPARTMENTALIZED, TRANSACTIONAL, NOT in a highly COMPETITIVE field, have some INTERACTIVE data center support, have a relatively small number of KEY PEOPLE targeted to use the systems, and limit usage to that which DOES NOT INVOLVE REGULATED TASKS OR FUNCTIONS.

• Integrated microcomputer strategy. A strategy of purchasing microcomputers which are designed to tightly integrate with each other and with the business data center is indicated for organizations that have a FREE-FORM management style, are INFORMATIONAL in their use of data, are highly INTERACTIVE in data center support, have a relatively large number of KEY PEOPLE who would expect to have a small system, and whose use of the systems may involve data reported as a part of the government REGULATION OF BUSINESS ACTIVITY.

• Prohibition strategy. Some businesses should not permit the employment of microcomputers at their current state of development. These are firms whose business is FREE-FORM, who have no defined information strategy, who have either a BATCH data center or no central data processing organization, and who would expect to make computer facilities available to nearly all the administrative, technical, and professional personnel.

While it may seem difficult to decide overall microcomputer policies without considering a single application, making such decisions on an application basis in the absence of overall policy results in getting too close to the immediate problem to see the full, overall business implications. In a corporation, microcomputers are corporate resources whether they are purchased as such or not. As corporate resources they must first serve as an instrument of corporate policy. If your business fits the description of a "prohibition strategy" it does not mean that you can never use microcomputers; only that there are steps in defining your business information management and processing strategy which must be performed first. On the other hand, the best interests of the business may be served if the president has a terminal to the corporate data center on his or her desk instead of a microcomputer.

Once a basic policy has been defined it will be necessary to establish a set of guidelines for the evaluation of specific applications. Unless the data processing organization is to support each separate project, creating the kind of bottleneck which the microcomputer was supposed to alleviate, user organizations will have to do a certain amount of local analysis and justification with a minimum of support. A very specific set of procedures for formally requesting authorization to purchase a system is the best way to lead the user through such a process. Such a document should require the following:

• Identify the need. What is the problem or opportunity which has caused the microcomputer to be considered? This should not be a justification of WHY a micro would help, but a statement of why help is needed. For example, a need might be a system to manage the inventory of surplus furniture and equipment created by a change in office style, permitting the sale of redundant equipment through brokers.

• Define the time frame. A critical question in any potential application is the time within which it must be developed. This should be based on business factors and not on personal preferences. Be sure to start at the EARLIEST point at which the system would be useful as well as the latest point where it would still be of value.

• Define the benefit. Not all projects must be justified by a cost savings or an additional profit, but those which cannot must be evaluated carefully. If a project can result in a financial benefit, it should be estimated and its source identified. Beware of internal transfer payment savings; a department may be able to save $20,000 in data center charges by buying a $10,000 computer, but the corporation already owns the data center and cannot give that department's share back. The benefit should not be presented as a full cost/benefits analysis including conformance to corporate objectives for return on investment at the early stage—the time required would not be justified.

• Describe the application. It is important to know what the proposed system will do and how it will interact with the rest of the business. Here are some key points:

- What is to be done? This should be stated in functional, user-oriented terms and not in technical terms.
- What will be the period of use, both in terms of total life cycle and in terms of frequency of use?
- Who will use the system and its data, both within the target organization and without?
- How much information is to be collected, per use and in total?
- How much daily work is involved and how is that estimate derived?
- What is the source of the data for the application? Is any of it already on a computer database?

• Why should a microcomputer be considered? The first choice of computer power should always be the data center. If a microcomputer is to be used it must be justified by the lack of support for the task in the data center, excessive development time, excessive cost, or some other specific reason. It should be the requesting user's responsibility to answer this question because that policy deters frivolous requests.

• What are the organizational implications? Will additional personnel, permanent or temporary, be required for the job? How will the people be trained, during working hours or overtime? Will job level upgrades be necessary to retain personnel once they learn new skills?

• What specific interactions with other systems are required? If the application must exchange information with another computer system, that system and the form of exchange must be identified. If the microcomputer must attach to a mainframe as a terminal, for example, both the micro and the mainframe must support this.

Applications in a form similar to that above can be submitted to a review committee for preliminary screening, and turned over to a staff for detailed exploration if desired. If possible, some guidelines for automatic approval should be considered in large organizations, perhaps based on conformance to a standard application profile or mode of use. For example, it may be desirable to make a microcomputer available to all professional employees of a certain type or all management employees at a certain level. This type of blanket authority recognizes similarities in organizational need and assumes that detail justification in 1 area is transferrable to similar areas with a minimum approval cycle. Whatever form authorization may take, there is a special problem with small computer systems which must be addressed very specifically: capacity.

VOLUMES & CAPACITIES—A KEY RISK

Most department heads or local users evaluating microcomputers have little exposure to the technical details of the systems and therefore no feel for what constitutes a practical assignment for such a system. There are several volume and capacity issues. First, the computer is normally called upon to store information for processing. Not only must the information itself be stored in some way, the instructions given the computer on how to process the information (the "program") must also be stored so that it can be

Microcomputer Buyers Guide
Personal/Microcomputers Used In A Corporate Environment

"run" as required. There is a limit to the amount of data which can be stored on a computer, and technical literature tends to specify it in units of "characters" or "bytes." Second, there is a limit to the ability of a computer to accept or display information in any form, via a terminal, printer, or communication line. Finally, there is a learning curve associated with the use of a new system and the personnel involved will not be fully productive initially. This can produce significant backlogs unless learning is done off-line, in which case overtime charges may be incurred.

INFORMATION STORAGE capacity of a computer is normally stated in "bytes," which means data characters, including letters, numbers, spaces, and special symbols. Nearly all of the microcomputers available use magnetic disks for storage, either in the popular semi-rigid envelopes often seen on word processors and called "floppy disks" or in fixed packages which are sometimes built into the computer itself and called "hard disks" or "Winchester disks." Often a vendor will say something like, "You have 320 "K" of data storage," where the letter "K" means 1,000 bytes. To make this information a little more digestible, 1 "K" of data is about 1-1/2 pages of double-spaced text, 6 business name and address labels, 2 average payroll histories for employees, or the accounting record for an active supplier or client. It is also about one thirtieth (1/30) of an average program. In general, a computer will require all of its information on a single application to be available at the same time if that application is being run. Thus, if you are planning to run a payroll for about 200 employees you will probably need about 100K of disk storage on the computer in addition to that needed for programs. This type of storage is usually called "online" because it is available at all times. Computers may also have data which is AVAILABLE but not online. Available storage is usually kept with the computer and consists of disks which hold the information for applications not being run. It is usually a good policy not to mix data from several applications on the same disk. Another form of storage is ARCHIVE storage, where backup copies of important data are kept. Archives are important because it is easy to damage a disk, and the data on a damaged disk is not usually recoverable.

Data on disks is organized into "files" or "databases" which are the collection of related information needed by a program to operate. A disk file is not unlike a manual file, it is sequenced by some piece of information such as name and it contains much the same kind of data for each element or folder in it. A single element of a file, is called a "record" and is analogous to the folder in a manual file. A unique key may be associated with a record or the record may be identified by its place within the file as a whole. Text files such as reports may treat each line as a "record" but not identify the line except as part of a sequence making up the report. Payroll or accounting files normally define a unique key, such as social security number, which identifies a record because each record has its own unique identity. Records may be further subdivided into "fields" which are pieces of information of consistent content. A personnel record might thus have an "employee number" field, a "name" field, and so forth. In a given file, all records will normally have the same field structure.

Estimating the amount of storage required for an application is best done by consulting manufacturer's specifications as covered later in this report. A preliminary estimate can be made by listing each piece of information ("field") which must be collected per "record" (per individual, or company, or client, etc) and estimating the maximum length of data which each "field" might contain. These are then added up to get a "record" length and multiplied by the number of "records" (employees, accounts, etc) to get the size of the "file."

DATA ENTRY on the computer is normally measured in keystrokes per hour, a quantity which has all the exactitude of bytes of storage and the same average comprehensibility. A champion production keyer working in an environment where the computer is never slowing the entry rate is capable of as much as 20,000 keystrokes per hour, about 5 characters per second. Data rates of 8,000 keystrokes per hour can be expected of the average worker if properly trained. If the the application requires more than just production keying from dependably formatted source documents the rates must be adjusted to include think time and document handling. A pilot test of an application can often be conducted on a word processor or typewriter by asking the operator to just type a list of the data which would normally be entered. Pilots will also help plot the rate of data entry as experience grows, giving an estimate of the learning curve and the start-up backlog which can be expected.

Data rates required for a given application are often difficult to estimate. The pilot concepts may be of a help, as many visits to other organizations or companies who are performing similar tasks. The amount of data to be entered in an application and the expected data rate will give a measure of the utilization of the equipment, an important consideration. Many microcomputers will not sustain 4 to 8 hours of constant keying per day because the keyboards are not rated for continuous service. Another impact of a high "duty cycle" is the importance of operator comfort considerations, the so-called "ergonometric" features of a system such as a tiltable, swivelable screen and detachable keyboard.

Both the data entry requirements and the data storage requirements of an application can be affected if some of the information which is intended to be processed by the application is already stored on the data center computer or on some other system. In this case, technical provisions to get the information onto the microcomputer will be needed but the local storage and data entry tasks may be reduced. The use of off-system data has significant advantages in a microcomputer application, but it also has its pitfalls.

REDUNDANT DATA is information which is stored in much the same form in more than 1 location. In some cases redundancy is valuable in that it eliminates references to many sources of information to get a few key facts about a topic. Personnel and payroll systems are often supported by different files of data and yet it is likely that the employee name and number will appear on both. This redundancy makes it unnecessary to refer to the personnel file for inquires about salary or payroll issues where the name of the employee is required. But redundancy on a microcomputer can waste local storage space, and information redundantly stored on microcomputers and mainframes can get out of phase, causing serious problems.

The difficulties of redundant storage are best understood by example. If a company has a large mainframe where expense account information is stored after accounting department reconciliation and payment and several microcomputer systems which are used by sales personnel to maintain their own expense records, a large amount of data will be stored in both locations BUT CHANGES TO THE DATA IN ONE LOCATION MAY NOT BE REFLECTED IN DATA STORED IN THE OTHER. Thus, if a historical change is made in the microcomputer database it will not necessarily update the mainframe, and an audit may uncover some uncomfortable variances. The ideal solution to redundant storage synchronization is to make 1 single file in 1 single system the change point for each piece of information and use the record of that change to drive a manual or automatic procedure to update the same information in the other places it exists. Where microcomputers are being used to extract and process information from a mainframe system, this means letting any CHANGES to the data be made on the mainframe and having the microcomputer periodically re-extract the files needed rather than to maintain its own copy. NEVER LET A FILE BE MAINTAINED ON BOTH A MICROCOMPUTER AND THE DATA CENTER COMPUTERS, OR SEPARATELY ON 2 MICROCOMPUTERS. Eventual loss of synchronization between the various copies of the file is inevitable.

The issues of volumes and capacities can be vital in the decision to permit the application of microcomputers. In the first place, high-volume, high-data-capacity applications have no business on small-business quality systems and should be handled by the data center computer. Where volumes and capacities are more moderate, it still may be advisable to consider the microcomputer offering of a manufacturer who offers traditional mainframe service policies with on-site repair and guaranteed response times. Such systems are more expensive than the small business and personal systems but offer hardware rated for continuous service and support adequate for applications where reliability is a key issue.

Growth potential must be considered in any application as well. Most businesses will expect a computer to have a useful life of at

Microcomputer Buyers Guide
Personal/Microcomputers Used In A Corporate Environment

least 3 years, with 5 years the most common period of depreciation. Because of the rapid progress in the technology of small systems a business should expect that a significantly more powerful and flexible system will be available for the same cost within 3 years. A major manufacturer of microcomputers estimates that the cost/performance relationships of small computers are improving at a rate in excess of 20% per year; other industry observers feel that this figure may be accelerated to 50%. If the current application outgrows the system in less than 3 years, it may be difficult to justify an incremental expense to adapt the nearly obsolete equipment to another application. GIVEN THE RATE OF ADVANCE IN THE POWER AND PERFORMANCE OF MICROCOMPUTERS, APPLICATIONS WHICH CANNOT COST-JUSTIFY WITHIN 3 YEARS PROBABLY SHOULD NOT BE SUPPORTED ON NEW MICROCOMPUTER HARDWARE.

With the discussion of volumes and capacities the evaluation of microcomputers leaves the policy and philosophy level and becomes necessarily more technical, but these technical issues retain a concrete business focus to the large user. Since individual applications cannot be permitted to develop independently and out of control, the equipment selected must fit the majority of PROBABLE APPLICATIONS within the business. This broad charter tends to favor systems from sound suppliers, solidly designed and engineered general-purpose systems, rather than system possessed of radical and untested concepts. Finding the right type of system depends on discarding specific application or technical orientations and beginning the evaluation with a broad perspective—the system view.

MICROCOMPUTERS—THE SYSTEM VIEW

A microcomputer is a computer, and as such has the same basic structure as any larger minicomputer or mainframe. The heart of a system is the central processing unit, containing the actual computer (usually a single chip—the microprocessor) and the memory used for program execution. Attached to the CPU, as it is normally called, are devices to store information such as disks and tape units and other devices (often called "peripherals") which, together with the keyboard/screen, link the computer to the human world. These peripherals may be capable of acting as input and output devices, such as the popular CRT terminals, or may only provide output in hard-copy form such as printers or plotters.

CENTRAL PROCESSING UNITS are often called "computers" themselves, since the actual computer power resides there. The CPU is normally quite small—often a single circuit board the size of a sheet of typing paper. On the board is the actual microprocessor chip which provides the computer with the ability to execute program instructions and perform the tasks which make up the application. The CPU operates on "instructions" which are grouped to perform a specific task and called a "program." The exact form of instructions depends on the type of microprocessor used in the computer. There are a wide variety of different microprocessor chips, as shown in Figure 3. It is popular to categorize microprocessor chips by the size of the binary number which they can read to or write from memory. An "8-bit" system is thus one which can access an 8-bit value in 1 memory operation. The newer 16-bit systems access double the information in a single memory use, making them potentially twice as fast. In reality there are many other factors which affect system performance at least as important as the 8/16 bit issue. Computers which have the same microprocessor chip have a degree of primitive compatibility, but may not be entirely compatible. Apple, Atari, and Commodore all produce computers based on the 6502 microprocessor chip but are generally not compatible systems. But FULL compatibility cannot be provided if 2 systems do not have the same microprocessor.

Attached to the CPU, sometimes on the same circuit board, is the MEMORY which the computer uses to hold the program which is "running" (the one which the CPU is actually working on) and the data which that program is using at that point in time. The amount of memory available for a computer depends partly on cost considerations and partly on the capability of the microprocessor chip. Memory size may also be set by the program which is runnning; having more memory than a program will use may not

8-BIT
6502 (ROCKWELL, COMMODORE) **6800 (MOTOROLA)** **6809 (MOTOROLA)** **8080 (INTEL)** **8085 (INTEL)** **8088 (INTEL) (16-BIT WITH 8-BIT DATA PATH)** **Z80 (ZILOG)** **NSC 800 (NATIONAL)**
16-BIT
8086 (INTEL) **Z-8000 (ZILOG)** **68000 (MOTOROLA) (32-BIT WITH 16-BIT DATA PATH)** **9900 (TI)**

MICROPROCESSOR CHIPS

Figure 3

result in any benefits and may increase the cost of the system. Most computers use memories of at least 48,000 bytes for business and professional applications. Memory is NOT used to store information for long periods. Most computers use a form of memory which loses its contents when the computer is turned off or when power is lost.

Combined with the microcomputer chip and the memory in a CPU may be units called "controllers." These allow the CPU to access disks and other storage devices or terminals, printers, and other peripherals. Without a controller, a device cannot be attached to or used by a computer. Controllers are often sold separately from the CPU so that users of a system who do not intend to use a specific device need not pay for the ability to support it. This provides cost advantages initially but may result in considerable incremental costs when an application is added to a system and requires a new type of external device. It may also result in the accidental ordering of a computer system which will not support the application for which it is inteded because it lacks a controller.

The hardware system has a parallel partner in the operation of the computer as a business tool: the software system. "Software" is a term given by computer vendors and users to the programs which are supplied with or used on the system to make it capable of useful work. A microcomputer with no software at all will perform no useful function except possibly heat the work space. The heart of the software system is the "operating system" or "executive." This is a program which is normally supplied by the manufacturer of the computer and is used to control the internal operation and information flow of the computer. The operating system makes it possible for programs to interact with the memory, storage devices, and peripheral devices of the system without having detailed knowledge of their operation. One key function of an operating system is the loading from appropriate storage devices of the programs which will perform the user tasks. A second function is to provide what is called "high-level support" for very complex and device-dependent functions such as reading or writing data from an external device. For example, an operating system might allow a program to request "read disk record," an action which involves positioning the read-write head of the disk, waiting for proper timing signals, reading 1 character at a time from the disk media into memory, and checking to see if any errors were detected. Without the operating system, each program would have to perform the detailed operations described and programming costs would rise alarmingly.

The programs which a computer actually runs must consist of "instructions" which mean something to the microprocessor chip used. This "machine language" is complex and difficult to learn, and programming directly in it is unusual. Instead, a series of "programming languages" have been developed which are more

Microcomputer Buyers Guide
Personal/Microcomputers Used In A Corporate Environment

closely allied to human languages and thus more easily used for development of programs. A programming language requires another program, called a "compiler," "interpreter," or "language processor" to translate the human-produced data into something the microprocessor chip can execute as instructions. Programmers, known for their tendency toward individualism, are often addicted to a specific language, and there are many to choose from. Not all languages, however, are available on all computers. Often a language is developed for a specific purpose; FORTRAN, for example, was developed for computerized FORmula TRANslation. To a programmer, the choice of a language may be the key issue in program development. To a user, the programming language used for a specific program which has been purchased for a user task may affect the performance of the system and the ease with which it can be moved from 1 computer to another.

A recent development in languages is the so-called "programmerless" language. This is a language developed for use by people not trained in programming, but whose use is likely to require at least some understanding of programming concepts. Many programmerless languages are designed for a specific purpose, such as the generation of reports from data files built by actual programs. Languages with even more specific functions may fall into the category of "general-purpose programs," designed to give users the ability to "tune" a basic function such as file building or financial spread-sheets to their specific needs.

Programs written to do a specific thing are called "application programs." Most users of microcomputers find it easier and less expensive to purchase programs for a specific task such as inventory management, word processing, or accounting than to have such programs custom-developed. The cost of a program and its difficulty are almost entirely a function of the job which the program must do rather than the size of the system on which it must be done. A payroll application on a large IBM mainfame would probably cost tens of thousands of dollars to develop, and the same program on a microcomputer is not going to cost less. In fact, because the limited features of most microcomputer operating systems and programming languages, microcomputer application development is often more expensive than that of large computers.

The interaction of the elements of the software system are shown in Figure 4. In this example, a program in the popular BASIC language is being executed to produce accounting reports. Each BASIC instruction is decoded by the BASIC INTERPRETER to a series of machine instructions. When an external service such as disk input or printer output is required, the BASIC INTERPRETER invokes an operating system request (often called a "function call" or "supervisor call") for the service. The operating system, a program in machine language itself, manages the memory, storage, and peripherals of the computer system. The hardware making up the system provides the environment in which the programs making up the software system execute, but since the instructions in the programs determine what the microcomputer actually does, THE SYSTEM'S INTERACTION WITH THE OPERATOR IS DETERMINED BY THE SOFTWARE.

USER	BASIC PROGRAM	BASIC INTERPRETER	OPERATING SYSTEM
"ADD 1 TO THE DAY OF MONTH"	"D = D + 1"	LD, A, (VARD) INC A LD (VARD), A	

"WRITE DAY ON REPORT"	"PRINT D"	LD HL, (VARD)
		CALL XLATE
		LD DE, RESULT
		LD C, 5H
		CALL OPSYS
			(PRINTS)

SOFTWARE INTERACTION

Figure 4

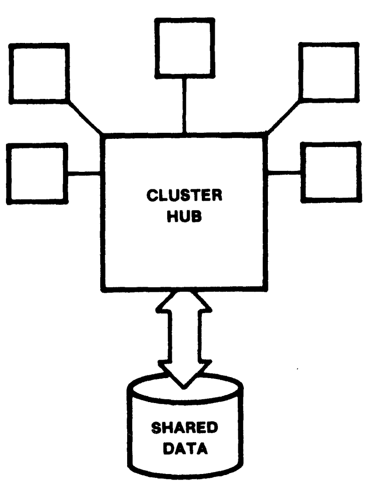
Software also defines the limits of the microcomputer's ability to integrate with other microcomputers or with the larger systems. The format of the data on the disks used for storage is determined by the OPERATING SYSTEM, which performs the reading and writing as a service to the user, and by the DISK CONTROLLER and DISK DRIVES. Two identical microcomputers can obviously read the same disks and are thus able to exchange data and run each other's programs. If the systems are NOT identical but have the same operating system and compatible disk drives and controllers, they can also exchange disk data directly. It is for this reason that users seek "standard" operating systems. Your IBM mainframes probably run one of IBM's standard operating systems, making data exchange with other IBM systems relatively easy.

The issues of compatibility arise at all levels within the hardware and software systems of a microcomputer and will be dealt with in the technical evaluations of each element, but there is an overall element of compatibility, SYSTEM COMPATIBILITY, which must be addressed. Each microcomputer system has a primary market for which it was designed and into which it is typically sold. Users who buy the system for its intended use are likely to be pleased with the results, but those who attempt to extend a system beyond its limits are not. Systems can be categorized as follows:

- Personal computers. These are systems designed to serve a single user in a home or limited business application. They are generally manufactured by companies who do not build equipment for large business use and are directed to the mass market. Issues of inter-system compatibility and integration with data center equipment are not normally addressed.
- Professional workstations. A step up from personal systems, these are primarily single-user systems built for business or professional use rather than for home applications. Often the professional workstation is the "low end" of a computer manufacturer's product line, and as such is generally designed for compatibility in some form to the rest of the line. Support for integration-oriented functions such as communication to data center equipment or other microcomputers is normally available.
- Small business systems. Designed for use by small businesses as their primary computer resource, these systems often support multiple simultaneous users, much like the mainframes in the data center. Having typically higher data and processing capacities than the workstation systems, they may be an attractive alternative for corporate purchase, but support for integration varies between vendors. Many manufacturers build such systems as the top of their product line and have little interest in helping the user migrate further or exchange data elsewhere.
- Cluster workstation systems. These computers are professional workstations with the added dimension of being linked together to form a "cluster" as shown in Figure 5. Data can be sent from 1 system to the other over the links, and files can be shared within the cluster.
- Distributed processors. The high end of the microcomputer line, these systems are very high-capacity and high-performance computers which are designed to operate as distributed processors in a network of computers, including as satellite computers to a mainframe. As Figure 6 shows, they differ from cluster workstations in that there is no limit to the distance spanned by the network, and in the fact that each distributed processor is a business system capable of supporting multiple users.

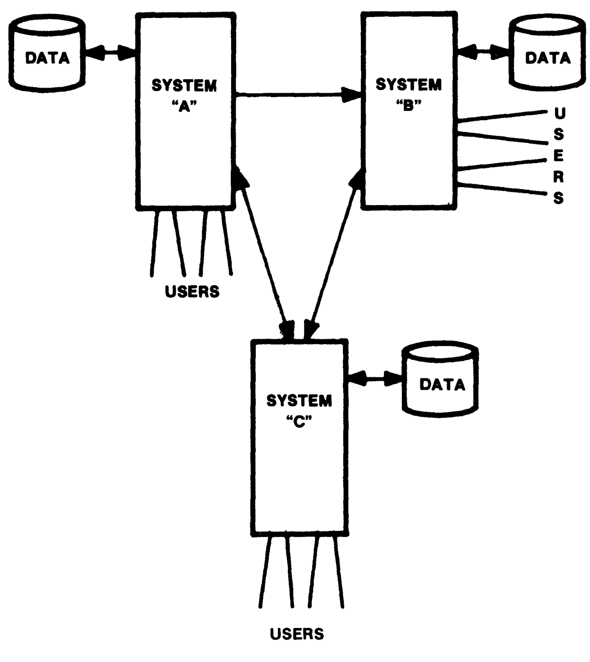
The system view should be the first step in the technical evaluation of microcomputer systems for corporate use, and it should include an analysis of each vendor's product line. Does the vendor supply a line limited to small business systems? If so, be alert to a tendency to employ non-standard and proprietary concepts designed to protect the market base. Does the vendor also sell large computers? If so, the systems may be a natural match for companies who use that vendor's mainframes and a very poor match for ones who do not. Whatever the case, the next step in corporate evalution of microcomputers is a detailed analysis of the elements of the hardware and software system,

Microcomputer Buyers Guide
Personal/Microcomputers Used In A Corporate Environment



A CLUSTER SYSTEM

Figure 5



DISTRIBUTED SYSTEM ARCHITECTURE

Figure 6

with particular attention to the issues most important to large businesses:

• Flexibility. Can the system perform in the wide variety of applications possible within a large company?

• Compatibility. Will the system interact as necessary with other computers or devices already in use?

• Standardization. Does the system employ standard concepts which will be easily learned and easily transported to other systems?

• Direction. Is the development of the system proceeding along a path which will continue to support the corporation's anticipated use?

SOFTWARE—WHY LOOK AT IT FIRST

We've already indicated that the functional capabilities and the operating characteristics of a computer are largely determined by the programs being run rather than by the equipment itself. The quantity and quality of software available for a system are thus key issues in the selection of a microcomputer. The corporate user in particular will want a wide variety of programs because the system's application in the corporation will probably cover a wider variety of tasks than would be normal in a small business. There are 4 criteria which software in general should meet:

• Availability. The more software available for a system the more likely the specific packages your business will need are available.

• Functionality. There are many different ways to do accounting systems or other common computer applications, most of which can be successful for a small business willing to make some adaptations in procedures. Corporations will probably be unable to adapt to the demands of a software package and so must select software which meets their probably more exacting requirements. Many small computer systems are designed for small-company use and will not.

• Transportability. A single computer with its software is a significant investment but a hundred of them can be a major committment. Software for that many systems could cost in excess of a hundred thousand dollars, and the ability to move that software to another computer if one is selected later may be the only way to protect the large investment. Not all software can be moved from system to system, and special expenses may be incurred in such a migration.

• Performance. Most microcomputers are powerful enough to out-perform their human operators in applications where constant human interaction is required. In more complex jobs which may require the searching of large amounts of data, however, the performance of the systems available will vary significantly. Corporate users often find it necessary to digest more financial data in a single ad hoc project than a small business would generate in an entire year of operation. This high volume requirement makes software performance a critical issue.

EVALUATION of the software available for various computers may be a difficult task because there is likely to be no single source of programs even for a single system, much less for the entire universe of potential purchases. The manufacturer of the system is of course the first contact in such an evaluation. Ask for a catalog of programs available from the vendor and for any lists of third-party programs which are known to be compatible with the system. Don't expect too much from this step—microcomputer manufacturers are rarely the major source of software for their own systems.

A second source of software availability is the microcomputer trade publications. Many popular microcomputers will have publications dedicated to them, containing advertisements of software suppliers. General publications such as Byte and Infoworld are very helpful in areas such as side-by-side comparisons of software for various systems. A simple software ad count will provide a measure of the amount of software available for the systems being considered, but very little on the actual capabilities of the packages. This deficiency can be overcome by using the reader service cards and obtaining more detailed data on a sampling of packages which appear interesting. Watch particularly for ads from large "software brokers" who sell hundreds of different packages for many computers. Often a quick scan of the ads of several of these suppliers will provide a measure of the relative amount of software available for the microcomputers you are evaluating.

A final source of software information is other users. Many microcomputers, especially those often used in business, have local user groups in major cities. If contacted during an evaluation, these groups can provide not only information on

Microcomputer Buyers Guide
Personal/Microcomputers Used In A Corporate Environment

software availability but practical experiences in using the systems. Be prepared for a certain degree of parochialism in such groups—the hard core dissidents and dissatisfied purchasers will rarely maintain active participation and those who do are likely to regard their selection less critically than you might like. If you are going to visit user groups at all, visit 1 for each vendor to round out the accolades.

SOFTWARE COMPATIBILITY with the data center computers or with other computers within the corporation may be a factor in the selection of corporate equipment. Understanding compatibility issues requires an understanding of the way in which software is provided. Any application packages are programs themselves, written in some programming language. A program written for one computer may be run on another if the following conditions are true:

• The language is compatible. If the packages are provided in machine language only (which is often the case), the computers must be OBJECT-CODE COMPATIBLE. If they are not built by the same vendor, a benchmark test to prove compatibility is MANDATORY in this case. If the package is available in the language the programmers wrote it in, it is said to be provided in SOURCE CODE form. Source code programs can be run on other computers IF A LANGUAGE PROCESSOR FOR THE LANGUAGE USED IS AVAILABLE. Language compatibility issues will be discussed later in this report.

• The program can be transferred to the other system. You cannot run a program which cannot be loaded onto the computer system. If the 2 systems have compatible storage media, such as floppy disks, the programs can be transferred easily by moving the disk from 1 system to the other. If not, it may be necessary to send the program from 1 system to another over a data communication link or have a service organization copy the program to a compatible form.

• Resources to run the program exist on the other system. An accounting system written for a large IBM mainframe is unlikely to run on a microcomputer, even if the languages are compatible and the program can be transferred, because there would not be enough memory or external data storage to support it.

A second type of compatibility between microcomputer software and software on mainframe systems is the ability to exchange information. A few microcomputer/mainframe combinations can exchange data on storage media such as disks or magnetic tape, but most such interfaces are via data communication—the transmission of data from one system over a private cable or phone connection for receipt by the other. In order for a communication connection to be successful, both the microcomputer and the mainframe must have software support for it. Text files can be sent from 1 system to another by having the microcomputer emulate a terminal device which is supported on the mainframe. Be sure that you check the type of terminals the mainframe can support, and evaluate all the terminal emulation packages available for the microcomputers. IBM users may often find that hardware devices called "3270 emulators" will provide microcomputers with the ability to attach to their mainframes as though they were part of the popular 3270 display system. IBM and other vendors also support the direct attachment of some terminals and some microcomputers may be able to emulate these. The IBM PC can emulate IBM's 3101 terminal, and the DEC Rainbow can emulate DEC's popular VT-100 terminal. Figure 7 shows micro-to-mainframe communication using terminal emulation.

Transferring data which is NOT plain text from one system to another is much more difficult. Many data files contain binary numbers which duplicate control characters used in the transmission of text, such as TAB, CARRIAGE RETURN, and LINE FEED. Since these characters are regarded as formatting information by most systems, they are often stripped out of the transmitted data stream, and their loss destroys the integrity of the non-text file. Some complex communication support packages for microcomputers and mainframes will support the transfer of non-text information, but such a capability is rare and likely to be expensive. Where the exchange of non-text files is a major factor YOU SHOULD REQUIRE A DETAILED EXPLANATION OF THE DATA EXCHANGE SUPPORTED BY THE MICROCOMPUTER, particularly insofar as it relates to your data center equipment.

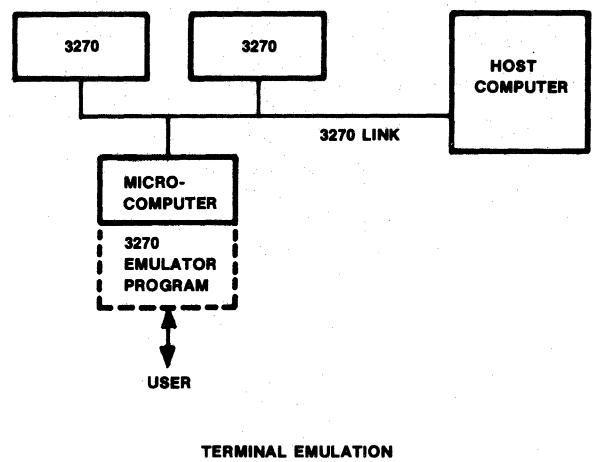


Figure 7

A final issue in software compatibility and data exchange relates to the transfer of data between microcomputers themselves, or between microcomputers and business word processor systems. Most microcomputers considered for corporate use will support word processing and other general functions. It may be desired to transfer a microcomputer data file created by a word processor or a popular financial spreadsheet package from one system to another. Such transfers are often impossible, not because the data cannot be exchanged but because the format of the information is not the same for both systems. One popular word processor, for example, marks the end of each line of text with a line feed character only if the line is not the last line of a paragraph and with both a carriage return and line feed if it is the end of a paragraph. Another uses a different structure. When one of these systems sends a data file to the other, the result cannot be formatted properly. THE ONLY POSSIBLE ASSURANCE OF DATA COMPATIBILITY BETWEEN TWO SOFTWARE PACKAGES IS THE SPECIFIC GUARANTEE THAT IT EXISTS. YOU CAN NEVER ASSUME OTHERWISE.

SOFTWARE IN THE CORPORATE VIEW may present unique problems. The average business package is a very small and very expensive module which can be easily transported and copied. You may not object to your employees copying their favorite packages for home use, but detection of this by the software manufacturer can result in a lawsuit which you will probably lose. If you have purchased a source license for the original programmer instructions so that the package can be tailored or moved to another system, you have probably executed a non-disclosure agreement which holds you responsible for the action of your employees with regard to unauthorized release. If one of your employees sells the proprietary programming information released to your corporation under such an agreement, the damages for which you are liable may be considerable. You may elect to risk the relatively small legal exposure where no specific licenses or non-disclosure agreements exist, but ANY PROGRAMS PURCHASED UNDER NON-DISCLOSURE AGREEMENT SHOULD BE HELD IN CONTROLLED ACCESS FACILITIES.

Operational and functional issues aside, software selection is made difficult by software's dependence on proper documentation and training for effective utilization. Unlike large packages for mainframes, microcomputer software is rarely supported by vendor training, so you will have to rely on either third-party training or the documentation. The more popular the package being considered, the more likely that training and tutorial manuals will be provided by developers who specialize in the application of other firms' products. Sometimes the availability of this material will outweigh functional advantages of less popular packages. A small corporation recently chose a popular financial spreadsheet program over a less-famous competitor even though the technical evaluation showed some clear advantages of the "brand X" package in the areas of intended use. Why? Because the more popular package had an extensive videotape training series available from a computer education firm and the other did not. Although it is diffucult to generalize on the relative value of features and less tangible issues such as documentation and training, remember that the best features are valueless if they cannot be understood by the operators.

There are some specific things to look for in documentation for microcomputer software:

- Multilevel manuals. A single manual for a complex package will tend to take either a reference approach or a tutorial approach. Either will be the wrong approach at some point. During training, an organized intoduction to the software along application lines is very useful, but as skills develop, it will become necessary to provide instructions along FEATURE lines. Beginners will say, "How do I enter totals?" while more experienced operators will ask questions like, "How do I tell the print command to print just this part of the form?"

- Proper setting of skill levels. Some tutorial manuals are an insult to the intelligence of a worker. Don't assume that your office staff, first line managers, and executives are illiterate just because they are not computer-experienced. When you look at a manual from the perspective of language style rather than technical content it should address the level of people who you will have reading it. Your executives will probably get little enjoyment from familiarity exercises like having the system type "HI! I'm your computer!" There is a tendency to package elementary computer concepts in grammar-school style. All it will do is put the readers to sleep.

- Illustrations. The best documents show pictures or drawings of what will appear on the screen. Being able to positively identify the point of reference is a big psychological boost for a tentative learner, who otherwise will waste time just looking around to see if this is the right section of the manual. Related to illustrations are examples; there should be samples of the operation at each step of program use. In very technical packages, be sure that an example of all the options and forms are given. Any form of programming language, including the generalized packages which provide you "programmerless" programming, should provide sample programs.

- Special instruction aids. Many software packages will come with "demo files," "sample programs," or other aids which are supplied on the program disk or tape with the package and which can be used by the operators to help build skills. These aids can be invaluable in getting the CONCEPT of the package across, normally the most difficult step in the learning process.

PURCHASE OF SOFTWARE often requires special attention. Once you are comfortable with the quantity of software available for a particular set of systems, you should consider the average cost of the packages and the issues of multiple system licensing. Any software package designed for accounting, word processing, data handling, or report generation which costs less than about $250 retail should be evaluated carefully. Quality software is expensive and only a very large customer base could justify a lower price for a full-feature product. You do not get bargains in software. Computer languages, excluding the BASIC usually provided with the systems, are likely to cost even more. Operating systems may range in cost from no charge to thousands of dollars, so exploration of their technical features will nearly always be necessary. These costs may be minor for a single system but will add up quickly when quantity corporate purchases are being considered. The average professional user, according to retail statistics, purchases over $800 worth of software packages for the computer. Some vendors will permit multisystem licenses at a lower cost, and more will provide quantity discounts on the purchase of large quantities of software. At quantity one hundred, you should be able to secure software at less than half the manufacturer's suggested price (NOT the prevailing retail price, which may be discounted 35% or more for quantity one). DON'T ASSUME THAT YOU HAVE THE RIGHT TO USE ONE COPY OF A PROGRAM ACROSS YOUR ENTIRE CORPORATE MICROCOMPUTER BASE. You almost surely do not, and software piracy is becoming a hot issue with many vendors. Don't become a test case.

Prices of software will vary considerably with sources of supply. One widely-used business package which retails for about $250 can be purchased in quantity from the vendor for $125. This discount can send a purchasing agent into a frenzy of self-congratulation until it is learned that mass software distributors will sell the same package in quantity for $69.95. Many microcomputer software vendors do not desire to deal with the end user in any form and will discount heavily to volume purchasers for resale. A quick scan through the ads in popular microcomputer magazines will help avoid paying more than necessary for a package. Don't take the quoted price for granted either; many large distributors will sell directly to quantity buyers at dealer prices.

When buying multiple copies of software, it is VITAL that you do not make the entire purchase at one time. Buy one copy of the system at whatever price you can get and check it out first, particularly if you are buying from a distributor. If you get the wrong thing it is better to have only 1 of them than 100, and return policies on software which has been opened are consistently unfavorable. Before you open any package, inspect it for damage and read the labels on the manual and disks. Is it the right package? Is it the version for the system you want?

A final issue in purchase procedures is the determination of policies on future upgrades to the software. Most programs are periodically revised to correct problems and to add new features. As a user, you will want the OPTION to go to a new version when it becomes available without paying for the software a second time. Find out what the upgrade policies are for the package, and find out how often updates take place. You will also want to know whether data files, etc, from previous versions of the package will still operate with the newer version. Compatibility with older versions of the same program is especially important in the type of software you will be using most of the time; application packages.

Little has been said up to now on the functionality to look for in software. Some of these issues will be covered in the sections on each software type, but functional capabilities are difficult to deal with in abstract. One suggestion from a major corporate purchaser is to select several applications to "pilot" the use of microcomputers in the business. This will provide specific requirements against which the capabilities of the systems can be measered. Often this type of focus will identify the critical requirements more easily than unstructured technical review. In general, corporate buyers should seek software which has a high degree of flexibility and which addresses the area of application from the perspective of a large business. A ledger package with 20 line items may satisfy a small business but will certainly fail in a corporate environment.

APPLICATION SOFTWARE—THE BIGGEST DECISION

Most of the work done on any computer you buy will.be done by software packages designed for a specific application—accounting, inventory, financial reporting, etc. The success of any corporate purchase of microcomputers is largely the success of fitting available application software to tasks within the business. Small users with well-defined needs often select a computer by selecting the application software and then purchasing hardware on which it will run.

FINDING THE BASIC THRUST of a package is probably the first step in evaluating it. Packages may be specific to a business application, designed for a particular computer-oriented task, or targeted for all the needs of a professional or manager. Within each category, packages make different trades in the balance of flexibility of application versus ease of use. The more specific a package is designed to meet a need, the less human effort is required to use the package in its target area and the narrower that target area is. For example, if an accounting system is designed specifically for medical practitioners the receivables, payables, and ledger programs can be set up to the medical environment, making the installation and use of the software very easy for a doctor. But a druggist might have more difficulty in using the system, and a lawyer might find it totally inapplicable. A package which provides the user with the ability to set up accounts, define billing formats and account for materials received and payed for in a general way may be useful to any business but will require more from all in order to be used.

Microcomputer Buyers Guide
Personal/Microcomputers Used In A Corporate Environment

Corporate environments, having many applications within a single company, will probably find that the purchase of the most general packages will be preferred. This will assure a wide applicability of a single piece of software, making quantity purchase and volume discount possible. It will also probably require some staff support for the setup of applications, since the use of these more generalized systems is always more complex.

The packages designed to perform a computer function such as report handling or data entry, storage, and retrieval are the most general of all application programs, and the most difficult to use. Popular "database" systems provide users a way to design their own data entry forms, accept data into files, locate information for update, and extract reports. In the proper hands, these systems are almost programming languages. Most large businesses find these packages indispensable, but their use should be monitored carefully.

The most recent trend in application software is the USER-specific package. Unlike the business package which attempts to answer the needs of a particular business function, this type of software is designed to fulfill ALL the needs of a type of user. Most often these packages include a basic financial spreadsheet program, a primitive word processor, and some data entry and report generation capability. This "userware" is designed to provide a worker with all the software capabilities needed through a single and consistent mode of operation, and may be the easiest to use of all software systems. Their problem is in extension—the need to go beyond the basic capabilities provided by the package for 1 or more tasks. Basic word processing may handle casual memos but will fail on a technical proposal. Limited financial spreadsheet programs will do expense accounts but may fall short on a critical project requiring large-scale financial trend analysis.

A pervasive feature in professional and business software today is the "menu" concept. People who use computers infrequently are often puzzled by the format of information which is expected at any point in the operator/system dialog. For example, the prompt "SELECT PROGRAM FUNCTION" is likely to send an operator thumbing through the manual, wasting time and getting frustrated. If that question is replaced by a "menu" of selections such as that shown in Figure 8, the operator has a clear idea of the possible actions and can make a choice without reference to other documentation. Menus, like anything, can be carried to extremes, particularly when the system is used often and the operator learns the procedures. When the selection of an action requires multiple selection in levels such as shown in Figure 9, the menu structure can become intrusive and wasteful. IBM, in their excellent user interface for their Information Network, provides the user with the ability to enter the selection for multiple menus at a single entry. Referencing Figure 9, this would mean that at the SELECT FUNCTION menu an operator could view a personnel record by keying 1:2:1 (PERSONNEL SYSTEM: EMPLOYEE FILE: DISPLAY EMPLOYEE RECORD). Studies made of operators who do extensive computer data entry show that when formatted information is entered (such as the response menu selection requests) experienced operators do not even read the screen and resent the extra time required to enter these incremental commands. Programmers and data center operators also demonstrate this resentment of menus—feeling they are "condescending." Before selecting a system or software which is extensive in its use of menus, evaluate the pattern of usage and the ability to bypass the menu. MOST USERS QUICKLY PASS THE LEVEL OF EXPERIENCE WHERE EXTENSIVE MENUS ARE NEEDED.

PLEASE SELECT YOUR FUNCTION

1 - PERSONNEL
2 - ACCOUNTS PAYABLE
3 - ACCOUNTS RECEIVABLE

ENTER YOUR SELECTION [ ]

A MENU

Figure 8

SELECT FUNCTION:
1 PERSONNEL → **SELECT FILE:**

1 WAGE RATES
2 EMPLOYEE →
3 STATE TAX

SELECT ACTION

1 DISPLAY
2 UPDATE → **DO**

MULTIPLE LEVEL MENUS

Figure 9

At a recent meeting of a computer user group a discussion of the merits of menus provided an opportunity to compare the ease of learning and using a menu system versus a non-menu system. The computer manufacturer had just released a "professional workstation" computer which was menu-driven, a departure from all its previous products. The menu was overlayed on the previous structure so that a user could leave the menu structure at any time and interact with the system in the older way. Users of the new system reported that within a month they found the menus unnecessary, and by the time they had used the system 2 months nearly half were looking for a way to disable the menu function altogether. All the operators who learned on the menu system found that their ability to use the full capabilities of the system was impaired relative to those who learned on the non-menu version. It is generaly agreed that new computer users gain initial confidence in their use of a system through aids such as menus, but many quickly find an extensive menu structure as childish as a first-grade reading text appears to a high-school student.

THE FIRST ISSUE in application software review for a corporation is the type of applications which are likely to be supported by microcomputer systems. Large businesses normally have basic payroll, accounting, and inventory systems available on their data center equipment. It is unnecessary and actually potentially harmful to allow these functions to be duplicated on microcomputers. The "average" corporate user will probably find the following applications most useful:

- Word processing. More and more professional and administrative personnel are finding that producing draft reports and other documents on word processors rather than handwriting them saves time and results in better organized material. As keyboard/terminal use increases among professional users the willingness to key material also increases.
- Financial spreadsheet applications. VisiCalc and its competitors are probably the most widely used software packages among managers and executives. These packages let their user define a financial spreadsheet similar to that shown in Figure 10 and to enter detail information for automatic calculation, totaling, and crossfooting. Spreadsheet packages are actually "programmerless languages" in that a user must define what the spreadsheet will look like and what the entry relationships will be, but the structure of these commands is very friendly and easily learned. In some cases, predefined "templates" of instructions can be purchased for specific functions. The versatility of these packages is considerable; it is actually possible to write a primitive accounting system in one!
- Time management and scheduling applications. Many managers and salespersons find that a computer package to post appointments, find free time, and provide a daily list of scheduled events is very helpful. The only difficulty in using such packages is that you normally are running some OTHER program when you need to access your schedule. This package is often bundled into the "complete user package" discussed earlier, and for a user

Microcomputer Buyers Guide
Personal/Microcomputers Used In A Corporate Environment

	A	B	C
1			
2	SALE AMOUNT	PROFIT	COST OF GOODS
3			
4	$ 575	$125	$ 450
5	100	25	75
6	600	100	500
7	1,300	400	900
8	820	120	700
9	———	———	———
10	$3,395	$770	$2,625

A SPREAD SHEET

Figure 10

whose time spent in either word processing or spreadsheet work is not extensive it may be a worthwhile investment.

• Plotting, graphing, and other visual-impact packages. More and more managers are learning the value of a presentation or document with visuals for impact. Even when the material is for the executive or professional's own use, the ability to produce charts and curves from raw data often aids in the visualization of a trend or problem and thus contributes to a more timely and effective solution. The best packages operate as accessories to the spreadsheet programs already discussed and give the user the option to plot the results rather than to display them in conventional financial column form.

• Special-purpose packages. Professionals such as architects, engineers, actuaries, or others who must use highly mathematical methods occasionally may find that a package to support their specific needs for modeling, formula development, regression analysis or simulations may be invaluable. Such software is available for many microcomputers. The typical alternative to these packages is either outside timesharing or data center programs, and both should be carefully evaluated before capital investment in special hardware and software is made.

PROVIDING EXECUTIVES AND MANAGERS WITH COMMON TOOLS can be an asset in management development and can improve the quality of internal reports, forecasts, and presentations. If all the equipment and software used is common to all users, rather than a mixed and possibly incompatible conglomerate, specific types of reporting can be encouraged by providing "templates" and style guides to all involved personnel. Internal "user groups" can be encouraged to promote the exchange of application ideas, tips on usage, and solutions to common problems. But it can also encourage expansion of the use of microcomputers beyond the scope intended by the corporation.

MOST CORPORATE SOFTWARE TENDS TO BE GENERAL-PURPOSE rather than strictly single-use packages. This increases the level of skill needed to develop and support an application on the microcomputer because a basic form of programming is needed to set any job up for use. The requirement for programming skill tends to divide the new users into "haves" and "have-nots," with the "have-nots" getting little accomplished with their investment and constantly requiring staff or external support while the "haves" work to test their new expertise on applications which rightfully belong in the data center or for which no true benefit exists. Programming is an insidious process, luring otherwise conservative managers to tinker with the system for what may be claimed to be valid business reasons but may actually be just a fascination with the process. While no one would suggest controls to stifle the genuine creative use of microcomputers, the need to prevent frivolous tinkering is very real. Here are some guidelines:

• Require a justification for each new package purchased. A company buying hundreds of financial spreadsheet application programs may find the unit cost to be so low that they approve new users as a matter of course. The cost in lost time due to tinkering may far exceed the software cost.

• Monitor the productivity of new users. Learning curves may result in loss of productivity early in the use cycle, but consistent degradation of performance may indicate a waste of time in working on "new ideas" on the computer. Occasional visual checks may also be in order. If the first-line manager who gets a new system spends 5 hours a day working with it, ask yourself how operation without it was possible.

• Encourage experimentation after hours. Many companies make it possible for their personnel to buy systems from the company at the volume price and many finance them through payroll deduction. Others encourage morning or evening projects to expand computer literacy and try experiments in new application areas. If such creative energies are channeled into legitimate after-hour paths, they will surface less often during end-of-the-month processing.

CUSTOM PROGRAMMING of applications, either by external resources or by department personnel, is the most thorny issue of application software. Most users consistently underestimate the problems associated with writing new programs, and the number of professional microcomputer programmers available is limited. Most often a computer store will offer to produce a special package for a "reasonable" cost in order to sell equipment, or will recommend an "expert" to do the development. The expert often turns out to be a hobbyist rather than a professional, and the user may spend tens of thousands of dollars finding out that the task is impossible on the equipment selected. The legitimate reasons for custom programming reduce dramatically as the size of the business increases. For the average corporate user NO SPECIAL PROGRAMMING SHOULD EVER BE CONSIDERED EXCEPT THAT WHICH IS EXECUTED BY OR COORDINATED THROUGH THE DATA PROCESSING ORGANIZATION. In particular, programming through retail outlets should be avoided. If you must get something special developed, use only qualified system or software businesses who can supply references and demonstrate similar applications on the type of system being purchased. You may also wish to consider deferring the purchase of equipment until the program has been satisfactorily demonstrated. If the software writer has "extensive experience in the XYZ computer," how was it acquired unless the company has such a system? If they have it, let them use it for development and buy your own only WHEN THE PACKAGE WORKS TO YOUR SATISFACTION.

In-house programming, within the user organization rather than in the data processing department, is a similarly difficult issue. Some programming languages are relatively easy to learn, and many businesses have found their managers and professionals happily writing programs for every purpose from creating new games to developing departmental accounting systems. While these abuses are sometimes obvious and easily dealt with, often the department involved honestly feels that a valuable service is being performed.

That businesses should avoid having non-data processing personnel doing actual programming seems axiomatic. Most inflexible rules tend to be violated, and a practical business policy based on principles similar to those developed for the after-hour use of microcomputers is likely to be more successful than outright prohibition. Such a policy should be based on the following points:

• Programming languages should be made available only to user organizations who can demonstrate a need and who have personnel qualified to attempt programming. Sometimes control of the physical media containing the language processor is not sufficient—try having the copy center refuse to copy any computer manuals without data center authorization.

• Guidelines for the development of user-written applications should be established by the data processing group or standards committee. These guidelines MUST INCLUDE A REVIEW OF THE PROPOSED APPLICATION AND THE FINISHED PRODUCT BY DATA PROCESSING PERSONNEL.

• Non-programmers actually being used in part for programming tasks should be evaluated for possible transfer into the data processing organization if it appears that their primary interest is in programming. Some form of "exchange program" between the DP organization and the user may satisfy short-term needs of a

Microcomputer Buyers Guide
Personal/Microcomputers Used In A Corporate Environment

user to write programs. In any case, beware of placing excessive restrictions on genuine career interest—it will likely be satisfied elsewhere.

• Watch out for "key man" problems. Sometimes an employee will grasp at specially developed programs as a means of extending influence or increasing promotability. This can cause major problems if the desire is thwarted: "accidental" destruction of files or programs for example. Managers should be particularly cautious about encouraging employees to do programming. Without a proper organization for support and a career path for motivation, a "user programmer" can be a very dangerous employee indeed. WHATEVER YOU DO, DON'T USE PROGRAMMING OPPORTUNITY AS A CARROT FOR A PROBLEM EMPLOYEE. You will quickly learn the full meaning of the word "problem."

• Monitor the productivity of the next level of management in any local programming project. Often the greatest cost of such an effort is the loss of management focus due to preoccupation with the new programming function. Since the user organization is not familiar with programming problems, any difficulties may absorb full attention of key personnel. You may be able to devote 1 person to a programming task, but certainly not an entire department, and particularly not a line manager.

Software issues crystalize the problems with the use of microcomputers in any business. Their general review, as provided above, helps develop basic guidelines on the application of microcomputers. Specific and technical evaluation of microcomputers for corporate purchase should also begin with an evaluation of the software; operating system and languages.

OPERATING SYSTEMS—WHAT, WHY & WHICH?

Previous sections of this report have already identified the basic role of operating systems; the management of system resources in a way which simplifies and standardizes their use. Operating systems are programs, normally written by the computer manufacturer and sold with the computer itself. Below are the key elements that make up a typical operating system:

• Task scheduling. Computer activities, like human activities are often subdivided into tasks. Some tasks may be scheduled concurrently because they do not compete for resources, but when resource competition occurs, an arbitration mechanism is needed. The task scheduler of an operating system performs this function, forcing a competing request for a device such as a disk to wait until the device is free. While many people associate multiple tasks with multiple users, even single-user systems may employ multitask structures to permit such functions as printing simultaneous with normal system operation, a feature sometimes called "spooling."

• Device control. The number and complexity of program instructions needed to perform even a simple device function such as the writing of a character to a CRT screen is considerable. Operating systems provide a simplified form of device control by providing a series of "functions" which programs can access and through which devices are accessed. Some operating systems provide a degree of "device independence" in device control. This means that data being written by a program to the printer may be diverted to a temporary space on disk, for example, if the printer is unavailable. Device independence means that programs which might otherwise be unable to run can operate and display results later.

• File control. Operating systems on large computers provide users with multiple "access methods"; ways in which data can be read from and written to storage devices such as disks or tape units. When operating systems provide very sophisticated file control capability, programs running "under" such operating systems are simplified in the same manner in which device control simplifies programs. File control is the most complex part of any operating system, however, and most microcomputer operating systems provide only primitive file control capabilities, leaving more complex features to the programming languages or to the programmers themselves.

The operating system provides the basic environment within which all the programs and applications run. Although it rarely interacts directly with the user (an exception is the loading of a program, normally requested by the user directly to the operating system), it is the cornerstone of the software for any system. The reason for this is that the operating system "insulates" the programs being run from the specific hardware system. IF AN OPERATING SYSTEM WILL RUN ON TWO DIFFERENT COMPUTERS, THERE IS A HIGH PROBABILITY THAT PROGRAMS FROM ONE WILL RUN ON THE OTHER. This is because the programs see the system environment only in terms of the operating system's "function requests" and not directly.

Probably the best example of operating systems and transportability of programs is the success of Digital Research's CP/M operating system. CP/M, which stands for Control Program/Microcomputer, was developed for the Intel 8080 microprocessor chip. Its flexible design lead many computer manufacturers to customize it for their particular environment and sell it with their systems or to their users. This created a large customer base of CP/M users which induced software development firms to write more and more programs for the CP/M environment. The large CP/M program base became a major selling point, making more vendors and more users select CP/M systems. The result is that CP/M is the most widely used operating system on the market. Subject to media compatibility issues already covered, CP/M creates a compatible environment where it is run. A CP/M user can select a program from a mail-order company, identify the type of disk format required, and use the program on arrival with almost complete assurance that it will run properly.

Compatibility is not the only issue in operating systems, however. Many programmers who have learned on larger computers find CP/M and other similar systems to be excessively primitive and restrictive. As a result, other competitive systems have been designed which provide improved performance, greater flexibility in adding devices to the system, or just more advanced programming features. Further market fragmentation has occurred as the power of microprocessor chips has grown, bringing the newer systems to the point where some operating systems designed for minicomputers can be useful.

It might appear that the most sophisticated operating system available, having the most advanced features for management of resources and providing the highest level of "function request" service, would always be preferred. In fact, there is a trade-off between the complexity of the operating system and the resources the operating system itself consumes. Oeprating systems offering a high level of service do so by being larger and more complex programs, taking more of the computer's memory for themselves and leaving less to the application programs actually doing the user's work. Many complex operating systems are also slower in responding to user requests, a byproduct of the greater levels of service available. In applications where the advanced features of such a system cannot be utilized, the complex operating system can add to system costs and limit performance.

Microcomputer operating systems can be classified as follows:

• Vendor-specific operating systems. These are systems designed to operate on one particular make, and often model, of computer. Most often such systems are sold by the vendor with the equipment. They are normally single-user and offer the most primitive levels of "function request" support. For some types of computers in the personal/professional category, these are the only operating systems available. Examples of this type of system are Apple DOS or TRSDOS. These operating systems are by far the most common on microcomputers because of their extensive use in home computers.

• Generalized business/personal systems. These are designed by software developers for use on a group of microcomputers in a business or personal environment. Digital Research's CP/M is such a system. Unlike the vendor systems already described, such systems can be adapted to a wide variety of microcomputers. They are typically designed to support a single user and provide basic device and file control services. Their simplicity makes them inexpensive, easily used by non-DP-professionals and low in the memory and disk resources they consume.

• Commercial/industrial microcomputer systems. These are

Microcomputer Buyers Guide
Personal/Microcomputers Used In A Corporate Environment

normally provided by large microprocessor manufacturers such as Intel, Zilog, and Motorola. They are full-feature operating systems designed for a specific microprocessor and used by custom system developers as the basis for packaged commercial systems. The systems are often capable of supporting multiple users and complex devices and are expensive in terms of system resources required. Probably the best known example of this type of operating system is Intel's RMX-86 for their 16-bit 8086 product line.

• "Universal" operating systems. Bell Labs developed an operating system called UNIX for use on its own equipment, and has been licensing UNIX to other firms or most commonly to colleges and universities ever since. Recent policy changes, coupled with the deregulation/divestiture of AT&T, have led Bell Labs to reduce the license fee for UNIX, making it available to microcomputer users through their vendors at a practical charge. A flood of "UNIX-like" operating systems has also developed, and there is a dedicated cult of UNIX supporters. These systems are very popular in programming circles but may be excessively complex to end users of computers.

There are nearly as many operating systems as there are computers, and not all systems will be available for all computers. Nearly every vendor has a preferred operating system, and a corporation which has determined that it must purchase a specific system would do well to consider using the operating system preferred by the vendor. More and more vendors, however, are abandoning their own private operating systems in favor of adopting a generalized system whose base of available software is an asset to the vendor in that it eliminates the need to develop large numbers of private programs to make the system useful. There are 4 major contenders for the position of "de facto standard" in the world of operating systems:

• Digital Research CP/M for 8-bit computers. Also called CP/M-80, this product is the most widely used and widely copied of any operating system today. There are literally thousands of programs which can be run on CP/M-80 systems. CP/M-80 is designed for microcomputers using the Intel 8080 or 8085, the Zilog Z-80, or the National NSC-800. With special hardware support it will run on some of the Apple and Commodore systems and on the IBM PC. Although CP/M-80 is available from Digital Research directly, most users purchase it from their computer vendor to get a version customized for their specific hardware.

• Microsoft MS-DOS. IBM selected this operating system for its Personal Computer (as PC-DOS) and the brisk sale of that product has made MS-DOS the current leader in 16-bit operating systems. Nearly all of the Intel 8086 or 8088 microcomputers available will support MS-DOS, and many vendors supply it as their standard system. The software base for MS-DOS is growing daily, but has yet to reach the level of the packages available for CP/M-80. Like CP/M-80, the user normally acquires MS-DOS from the computer vendor rather than from its developer.

• Digital Research CP/M, 16-bit. The spectacular acceptance of CP/M-80 has so far eluded its newer relative for 16-bit processors. Digital Research made a version of CP/M-86 available for the IBM PC, but its high price discouraged sales to the point where it is estimated that less than 5 percent of the PC users have purchased it. Because CP/M-86 operates on Intel 8086 or 8088 computers which are not compatible at the machine instruction level with the earlier Intel, Zilog, or National products which support CP/M-80, it cannot fully take advantage of the software base for CP/M-80. A new version of CP/M for the Motorola 68000, called CP/M-68K, has just been introduced.

• Bell Labs UNIX and its derivatives. UNIX was originally designed for minicomputers which had more memory than the early microcomputers. It is not surprising that when the big-memory 16-bit systems became available, firms would begin to sell UNIX-type systems for them. UNIX has a considerable software base written in a programming language called "C" (as you might expect, there was a "B" earlier but apparently never an "A"), but not all UNIX-type systems support a "C" language compiler which has the same features as the minicomputer versions for which the program base was developed. UNIX and its family are popular primarily on systems based on Motorola's powerful 68000 processor. They offer a very friendly environment for programming and scientific/technical use, but many users find them too complex for general business or professional applications. Interest in UNIX is growing rapidly as license charges reduce and commercial software development for it increases. It is still hampered by the lack of a dominant data storage format standard for floppy disk, making the exchange of information between UNIX microcomputers more difficult and making the sale of package software a multiversion event for all developers who wish to address more than one vendor.

For a corporate selection process, the microcomputer operating system decision may be approached as follows:

• If a very wide base of software packages are considered necessary to the successful use of the systems, and if the projected use is of the professional workstation type, CP/M-80 may offer significant advantages. If you decide to use CP/M-80, your hardware selection process must focus on computers which support it. A word of caution. CP/M-80 is designed for the Intel, Zilog, and National microprocessors already mentioned. It will run on other computers IF A SPECIAL PIECE OF HARDWARE WHICH CONTAINS A CP/M-COMPATIBLE MICROPROCESSOR CHIP IS INSTALLED. Don't do that unless you already have a non-CP/M system and must upgrade it. If you want CP/M, buy what is DESIGNED to run it.

• If a wide variety of application packages is not as important because your corporate scope of use for microcomputers is limited, you may find products for MS-DOS or one of the 16-bit versions of CP/M which uniquely suit your requirements. This is particularly true for users who have requirements to perform tasks requiring large amounts of memory, such as scientific calculations or very large spreadsheets. Most users who buy MS-DOS do not do so for its leading position in the 16-bit world or for its superior features, however. They buy it because it is the standard operating system for the IBM PC. That fact is likely to result in an impressive growth rate in available software, and such growth will benefit all users of the system.

• If you have a primarily minicomputer-based data center which uses Bell's UNIX operating system or if you have a primarily technical or engineering application for microcomputers, you may want to consider UNIX systems. Many sophisticated mathematical, engineering, and technical programs are available for UNIX systems and some are even in the public domain, eliminating license fees. UNIX should also be considered if you expect to purchase a system supporting multiple users, a topic to be covered later in this report. If you are planning general business application of your systems, you may be better off with a business-style operating system.

• Unless you are buying a computer for a function like process control or another specific and technical application, don't even consider the big commercial operating systems provided by microprocessor vendors such as Intel, Zilog, and Motorola. Such systems are designed for users who will develop their own software, something corporate users should not consider doing except as part of the data processing activity.

If one of the above points seems to "jump out" at you in its applicability to your situation, consider making a specific decision on operating systems and using that decision to narrow the technical selection of microcomputers to those which will support it. If none seem particularly applicable, you probably do not have a great sensitivity to operating system selection and will have to use other criteria for evaluation and selection of candidate systems.

PROGRAMMING LANGUAGES & THEIR CORPORATE IMPACT

Although each microprocessor "chip" has a unique machine language which it is designed to support, programming directly in those languages is rare. Machine code is complex and difficult to learn, and the maintenance of programs already written in it is expensive. Most actual programming is done in languages designed for ease of use in specific applications, called "high-level" languages. A more primitive language, called "assembler" is also available for most computers, offering an improvement over direct use of machine code but fewer "friendly" features. Assembler is often used for complex programs

Microcomputer Buyers Guide
Personal/Microcomputers Used In A Corporate Environment

such as operating systems where the ultimate in programming efficiency is required.

Programs written in a language are said to be in "source code" form, and that form is readable and understandable by programmers directly. Source code is translated to "object code" or machine language by a form of program called a language processor. The translation can occur a single time after the program is written through the use of a "compiler" or on an instruction-by-instruction basis during each running of the program through an "interpreter," a program which decodes the source codes and performs the machine instructions. Compilers produce programs which typically run faster and use less memory, but are more difficult to program with because errors in the program are detected in a batch when the compiler is run rather than on an instruction-by-instruction basis during programming.

There are many program languages, each designed for some specific purpose. Some languages have been defined by standards organizations so that source code written for one computer can be moved to another and run with minimum problems, while others are defined slightly differently by each vendor. Such compatibility is called "source code compatibility" and requires only that there be language processors on the 2 computers which will accept and translate the same source code. Another form of compatibility, called "object code compatibility" means that the machine code produced on one system will run on the other WITHOUT REQUIRING A NEW SOURCE TRANSLATION ON THE TARGET COMPUTER. Object code compatibility requires that the microprocessor chip on both systems be compatible and that the operating systems be compatible, while source compatibility requires only compatible language processors.

Corporate users who intend to permit program development on a microcomputer should probably make any decisions relating to the programming languages based on recommendations of the corporate data center. If a standard language is employed by the corporation for large systems, it is desirable that microcomputer development also proceed in that language. Where several languages are used in the data center, selection of a microcomputer language is normally made based on issues of programmer familiarity with the languages rather than on their technical characteristics.

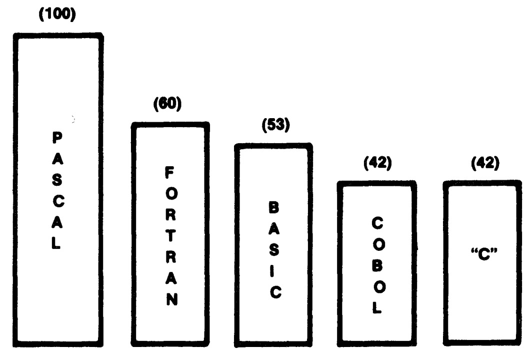
Since most corporations will probably not encourage user program development on microcomputers, the issues of programming languages to them will revolve around the availability of application software package. The languages which are supported by a given combination of microcomputer and operating system will affect application software availability as follows:

• Programs which are normally supplied in source code form, either from your own data center or from external sources, will require a compatible language processor to be useful. Many useful programs are available through user groups, government projects, or other public sources in source code form. Because the original programmer's instructions are available, changes to the programs to tailor them to your environment are more easily done than with object code programs (which should be regarded as non-changeable).

• Availability of a language processor for a given language will encourage software development companies who use that language to make object code versions of their programs available for the computer. If the language involved is highly popular and standardized, this can result in a considerable advantage in terms of available software.

Although there are hundreds of different programming languages, only about a half dozen are common enough to warrant attention by the corporate user, whose need to avoid the unusual and therefore uncontrollable disqualifies "minor" products. The relative popularity of these, as measured by the percentage of systems which offer support for them, are shown in Figure 11. The following are the languages you are most likely to encounter:

• BASIC. This is the easiest programming language to learn, and by far the most common language for microcomputers. Various attempts have been made to standardize BASIC but with little success. The closest approach to a standard presently found is the popular Microsoft form which serves as the basis for the BASICs of most personal computers. CP/M systems using Microsoft BASIC are generally completely intercompatible, but some changes in programs are needed to support migration between such systems as the IBM PC and the CP/M world. BASIC is available both as an interpreter and as a compiler. Professional programmers tend to avoid BASIC because it is unsophisticated and lacks some of the program structure aids which increase maintainability. The minicomputer BASIC popularized by Digital Equipment Corporation is compatible enough with Microsoft-form BASICs to permit practical conversions. Most BASIC systems use a form of "scientific notation" called floating-point arithmetic for their internal calculations. These systems are subject to a loss of accuracy when very large numbers are involved, so it is important to test BASIC versions where great numeric precision is required.



RELATIVE SUPPORT OF PROGRAMMING LANGUAGES

Figure 11

• Pascal. Developed as a language for teaching programming rather than as an actual commercial offering, Pascal is nevertheless gaining popularity among microcomputer users. Pascal has many features which aid in the production of well-structured and easily maintained programs, and recent efforts by the International Standards Organization to define a Pascal standard have been quite successful. There are 3 dominant forms of Pascal today, one being the ISO standard, one a version developed by the University of California, San Diego (UCSD Pascal), and the third being the original form developed by Nicholas Wirth. The forms are similar enough so that a programmer can work in one having experience in another, but not enough to make direct conversion easy. Pascal in some of its forms lacks good disk file support, and most versions have the same problems with large numbers as BASIC.

• FORTRAN. The oldest of the high-level languages, FORTRAN is popular primarily for engineering and scientific applications of microcomputers. The American National Standards Institute (ANSI) has several FORTRAN standards, the latest being the 1977 version. FORTRAN 77 programs are reasonably transportable, including compatibility between microcomputers and larger systems. Because FORTRAN has a scientific orientation which stresses large and complex calculations, it is often used on the 16-bit computers, and FORTRAN users should probably consider such systems seriously. Some FORTRAN compilers will support special hardware for scientific calculations such as array processors and high-speed mathematics processors.

• COBOL. Still the most widely used business programming language, COBOL is gaining popularity on microcomputers. Most microcomputer COBOL compilers will not support the full COBOL language as it is supported on larger systems, so users planning to transfer COBOL programs to microcomputers must study the differences carefully. Subject to this level of support, COBOL programs are highly transportable. Because the language is designed for business use, COBOL has good disk file

Microcomputer Buyers Guide
Personal/Microcomputers Used In A Corporate Environment

management capabilities, flexible formats for the display of dollar figures on reports, and high numeric precision for handling large numbers. Its disadvantage is primarily an old-fashioned program structure which many new programmers find restrictive and a tendency to require a lot of writing to get anything done—a characteristic which leads to it being described as a "verbose" language.

• "C." Like Pascal, "C" is a structured programming language, providing facilities which permit programmers to organize their programs for maximum development ease and maintainability. "C" was developed at Bell Labs and is associated with their UNIX operating system, itself written in "C." A great deal of government-sponsored development has been done in "C" and many of the programs are public domain, but most are of a technical nature. "C" is not yet a popular business language because of the usual problems with disk file and high-precision arithmetic support, but it is generally conceded to be better suited to high-performance applications than Pascal.

• PL/1. A very complex language developed by IBM in the 1960s, PL/1 provides what is probably the best combination of language structure features and business support available from any programming language today. It is also unfortunately almost frighteningly complex and difficult to learn. IBM had hoped that PL/1 would become the dominant programming language. It has not, and obviously will not. A subset of PL/1 called subset "G" has achieved some popularity on minicomputers, and Digital Research makes an excellent PL/1 compiler available under its CP/M operating system. A program written for that compiler has won a popular computer magazine benchmark performance test of programs for several years running. PL/1 should be considered a development-only language, however. Very few software packages are available in PL/1.

• ADA. The Department of Defense recently began promoting its own "all-purpose" language. Like PL/1, ADA has both business and structure advantages over the currently popular languages, but like PL/1 it is very complex—so much so that there is currently no microcomputer compiler for ADA which meets the DOD specifications. Several good compilers are available for functional subsets of ADA which may be useful to corporations who want to get ADA experience for future DOD contract work. Other businesses can ignore ADA at this point.

• Assembler language. This is not one language at all, but the collective name given to the most primitive form of language processor—one which provides only the basic ability to write in machine-code form. Programs written in assembler are the smallest and fastest, but require more time to develop and are most difficult to maintain. No corporate user should ever permit assembler to be used for program development outside the data processing organization. Most assembler programs are not source compatible with anything—the only limited exception is that of the Intel 8080 and 8085 assembler programs, which can be program-translated for the later Intel 8086 and 8088 microprocessors.

• Other languages. There are many other programming languages such as FORTH, LIST, APL, or RPG. These languages are supported to a greater or lesser degree on microcomputers, but are very specialized. Locating professional programmers for such languages may be difficult in many areas of the country, and program products for the languages are more likely to contain undetected errors because of relatively light usage. A corporation developing a high-technology product employed one of these less-common languages for development and encountered a major problem when a programmer dispute left them with no staff and unable to find replacements with skill in the proper language.

If you intend to permit program development on microcomputers, the language choice is much more important than if you elect to restrict such development to the data processing personnel. In some cases, it may be difficult to learn the identity of the programming language used for packages. Here are some guidelines for when to concern yourself with program language:

• When any form of custom changes are to be made on an application package, either by your corporation or on your behalf.

• When an application package you purchase is not widely used or not supported by a major software supplier.

• When the application for which the software is designed is subject to significant changes on short notice, perhaps due to changes in regulations governing the industry.

• When the scope of the application demands the ultimate in performance.

A SUMMARY SOFTWARE EVALUATION will for many users define their requirements sufficiently to make the selection of the hardware relatively simple. If you have developed any firm requirements with regard to application programs, operating systems, or programming languages, you should note them and apply them as the first selection criteria in reviewing microcomputer systems. Here are some tips:

• Many application packages will either specify a specific computer ("requires an IBM PC with 128K bytes of memory and 2 disk drives") or at least an operating system and basic configuration ("CP/M system with 56K bytes of memory and 500K bytes of disk storage"). Some will give several alternatives ("for Apple II, ATARI, or TRS-80 model 3 with 1 disk"). If you must have that package, you must have one of the target systems.

• Operating system selection will often narrow the computer selection down to a class of systems based on 1 or more microprocessor chips. If you want to use CP/M-80, you will need a system which is based on the Intel 8080 (obsolete and to be avoided) or 8085 microcomputers, the Zilog Z/80, or the National NSC-800.

• Language requirements may impact you in several ways. If a language processor is not available for a given system, programs for that system cannot use that language. If little use is made of an available language, you can expect it to be less powerful and more likely to contain hidden problems than that same language processor on a system where it is popular. BASIC is well-supported on all systems, but COBOL is commonly used only on CP/M or 16-bit systems. Another language issue derives from the common programming practices of a computer. Even if a language is available on a system and the language processor is well-designed and effective, the programmers for that system may not elect to use it to develop application programs. There is an excellent PL/1 compiler for CP/M systems but little software is written for it, so looking for applications written in it will waste your selection time.

Once you have identified the restrictions which the software evaluation will place on the overall system, you can begin to analyze the hardware issues. These are best taken in order of importance: system capacity issues, device support, expansion support, and flexibility.

MEMORY & STORAGE

Information on a computer must be in a form and location which the microprocessor chip can access in order to be useful. Things which the computer is working on at the moment must be stored in a way that the access is very fast and generally unrestricted, while those things which may be needed but can be "looked for" can be placed in less easily accessible storage. There are always more of the latter types of information than the former (How many simultaneous documents can you process at once?), so computers will always have more of the long-term "storage" (on floppy disks, for example) than short-term "memory."

Computer memory is the name given to the "work space" or "scratch pad" which the microcomputer uses to hold the program which is being run at the moment and the pieces of data which it is using. The operating system loads programs into memory for their execution, and running programs use the operating system to request that information on less accessible forms of storage, such as disks or tape, be loaded into memory for review or update. Memory on modern computers is made up of semiconductor chips, and its size is almost always given in the unit "K," or thousand. Because of the binary, power-of-two orientation of computers, a computer "K" is actually 1024 bytes or characters (2 to the tenth power).

Storage on microcomputers is usually in the form of disks, devices which record information magnetically on a special surface. Disk units may be of the fixed type, where the media of storage cannot be removed, or of the removable-media type. If the storage media

Microcomputer Buyers Guide
Personal/Microcomputers Used In A Corporate Environment

can be removed from the disk unit, it can be placed in an area not actually accessible to the computer until it is reloaded. This causes storage to be categorized as "online," meaning available to the system because it is loaded into a unit, or "off-line," meaning that it has been removed from the system.

Storage and memory interact in the running of a program. Programs and data are always maintained in storage until needed, then loaded and used. The transfer of information from storage into memory does not destroy its image on the storage device, so if a problem occurs, the image can be reloaded and used. Both storage and memory can be written to as well as read, so information not already available can be created and saved.

Memory is a quantitative issue in a field where many factors are very judgemental, so users tend to grasp at memory size of computers as a key factor because they can be compared. There are dangers to this, because just having memory doesn't mean that the microcomputer can actually USE it. The problem is an issue called "address space." If a computer memory contains, for example, 64K bytes of information, the computer must be able to access each one of those bytes, or "memory locations." It does so by means of an "address," which is just a number beginning with zero and increasing to the limit of the LARGEST NUMBER WHICH THE SYSTEM CAN ACTUALLY MANIPULATE. The total range of possible addresses for a computer is called its "address space." At any point in time, no computer can handle more memory than its address space. This is not to say that there are not ways in which to use memory beyond the limit of the address space—there are several. For example, each user or "task" can be given its own address space within a larger memory. This means that when the computer is doing work on behalf of task #1 it is using that task's portion of memory as its address space. The same task can also move its address space around as a "window" into a larger memory. Obviously, both of these techniques are anything but obvious and both will require special programming effort. The chances of that effort being made are pretty much dependent on the type of computer used and the popularity of large amounts of memory on that computer. Many popular CP/M systems will happily allow you to put as much as 512K bytes of memory onto them, but the address space of most CP/M-80 computers is 64K! Very few programs will ever be able to use that extra memory, and a single user will almost certainly not be able to buy even 1 application package which does. Nearly all 8-bit computers have an address space of 64K. The 16-bit systems have much larger address spaces, at least 512K, so memory restrictions with these computers are rare.

Another issue on memory is the amount actually used by the program. Most programs are written to require a minimum amount of memory, something users are aware of because the restriction is advertised. What is less known is the fact that most programs will also utilize only a certain amount of memory, even if more is available. Version 1.1 of IBM's BASIC program on their personal computer will utilize only 64K bytes of memory, even on a system with much more memory, unless special actions are taken.

How much memory is enough for corporate users? Here are some guidelines:

• On any 8-bit system, "enough" is all you can get up to 64K. Don't go over that unless the vendor can supply convincing demonstrations of programs which you need and which can use that increased memory for some benefit such as improved performance.

• On 16-bit systems, the operating system is normally more complex and takes more memory iteslf. 64K is a minimum on such computers, and most users will find that the practical minimum is 128K. Expansion to 256K is often useful, but beyond that, demonstrations of the value of further expansion should be required. For FORTRAN or scientific users, get as much memory as the programs will utilize—512K bytes or more is not uncommon on such systems.

• Word processing and spreadsheet programs are often able to use large amounts of memory. If your applications include these popular systems, consider expanding memory to the limit they will utilize if you prepare large documents or worksheets. The greater memory size with word processors will minimize the time it takes to move around in the document for changes and reviews. This report was prepared on a computer with 512K bytes of memory, and the word processor and document could both fit entirely in the memory of the system at the same time, eliminating the need to read parts of the document from disk each time changes were made. On professional proposals and technical reports where frequent backward reference and review is likely, this can improve productivity by 20 to 25%. With spreadsheet applications, the larger memory will allow more row/column combinations; larger spread sheets.

STORAGE DEVICE SELECTION is more complex than memory because there are issues beyond capacity which must be addressed. Storage of information on microcomputers is almost always on disk devices, tape units being used only for special purposes. There are 2 basic types of disk storage; hard and floppy. The hard disk units employ a fully rigid media with typically high storage capacities (5 million bytes or more) and a relatively high cost of media. Floppy disk storage uses the familiar flat, semi-rigid package found in most word processor systems. The media is inexpensive, but the storage capacity of the floppy disk is much less, about 1 million bytes as a maximum. Floppy disks are always removable, so many disks can be stored off-line and loaded into the DISK DRIVE when needed. Hard disks may be either removable like floppy disks, fixed, or a combination of the two. Fixed hard disks are usually of a generic type called the "Winchester," an encapsulated disk which is sealed to prevent contamination of the surface by dust, etc. Winchester disks have a typical capacity of 10 to 40 million bytes of data. Removable disks may be of a proprietary "cartridge" type or what is called a "Storage Module Drive" or SMD. The electrical connections for the SMD are an industry standard, so SMD disks will attach to many different types of computers. SMD disks are often very large, having capacities of 200 million bytes of data or more, and their cost is very high.

What type of disk storage is needed is a function of the application and the cost sensitivity. Some guidelines on the selection of disk type are:

• Applications which must retain large amounts of INDEPENDENT data such as documents, special project data, etc, will probably find that the floppy disk's ability to support removal and storage, coupled with the low cost of the media, make it the preferred choice.

• Applications which require the immediate access to very large amounts of data, such as an employee file, may find that the large capacity of hard disks presents the only way to store the file at all.

• High performance requirements, particularly where the application is searching for information through a data file, may indicate that the higher speed of the hard disks is needed.

• Unfavorable operating environments may cause damage to floppy disks because they are not protected from contamination. The encapsulated Winchester drives may be the best answer there.

• High security demands for an application may make floppy disks a poor choice because of their ease of concealment.

• Transfer of data by disk to other computers, of course, demands a form of removable media. Some companies attempt to move an SMD disk because of the standard interface, but the method of data storage on the device is not as standard as the electrical interface. It won't work in most cases.

Another issue in the selection of disks is the selection of the vendor. Most users buy disk storage from the supplier of the microcomputer as a packaged system, but many are attracted to the lower costs of the third-party disk suppliers. Corporate buyers should be very cautious in selecting any third-party disks. The disk drives are probably the most critical part of a computer system in reliability—they have the highest failure rate of the basic system, and their failure is usually catastrophic in that the system will not operate without them. Further, the disk drives and the operating system form a very close partnership. Although the suppliers may not advertise the fact, many add-on disk drives require modifications to the operating system to run. While these changes may be supplied with the disk and be made almost transparently to the user, THERE IS A REAL DANGER THAT THE

Microcomputer Buyers Guide
Personal/Microcomputers Used In A Corporate Environment

CHANGES WILL PRECLUDE YOUR MIGRATION TO NEWER VERSIONS OF THE OPERATING SYSTEM, at least until the disk vendor has made similar changes for that version. There is also a risk that the disk controller logic or the disk interface will interfere with the operation of some other element of the system because of a lack of knowledge on the part of the builder about the details of the electrical interface the microcomputer requires. Where there is a standard interface for these devices, often called a "bus standard," that risk is reduced.

FIXED DISKS provide freedom from contamination, simplify the use of the system by eliminating the insertion and removal of floppy disks, and provide higher rates of disk performance because they can operate with closer tolerances. But fixed disks aggrevate the most significant user problem with microcomputers—backup.

When a computer fails in any way, there is a chance that the information being currently written to disk will be in some way damaged, making at least that particular record and possibly the entire file useless. If the point of failure is the disk unit itself, the failure may cause damage to the recording media itself and make the entire disk unusable. All disk types are susceptible to this, but with floppy disks, copies of the data can be made easily if there are 2 disk drives on the computer. This backup procedure should be a part of the operational training of each computer user. With fixed disks, however, there is no way to remove the copy and store it in a safe place. The fixed disk user can thus lose EVERY PIECE OF DATA IN THE SYSTEM AND EVERY PROGRAM STORED unless that data has been copied to another device. The most common practice is to copy the disk to floppy disks for storage, but the transfer of a 10-million byte fixed disk to floppy disks of 500K-byte capacity will require 22 diskettes and may take several hours. Of course, not all the data may require backup, and "incremental backup" of hard disk to floppy disk is a practical procedure. If your files are very large, however, you may require a special device for backup, normally a tape unit of some type. Tape backup normally takes only about half an hour, but the units may be very expensive and are not available on all types of computers.

BACKUP POLICIES are very different from company to company, depending on the specific application requirements, but here are some general rules:

• Don't buy a system which has only fixed-disk storage, even if the manufacturer says they will load the programs from a special device for you. Get either 1 floppy disk drive or a form of tape for backup. You should also probably plan to always have a floppy disk installed for transfer of data to other systems or to load programs you purchase later.

• Very critical applications may need backup which can be performed rapidly. Some SMD disks are available with a fixed and a removable portion, and the fixed area can be transferred to the removable area for backup. NEVER ASSUME THAT JUST MAKING A COPY OF THE DATA ON THE SAME FIXED DEVICE IS BACKING IT UP. You just get 2 copies to lose if the media is destroyed. IT ISN'T BACKUP IF YOU CAN'T TAKE IT OFF THE SYSTEM.

• Don't skimp on the purchase of backup devices. You may find out that the copy is not reliable at a very bad time.

• Complex backup procedures should be avoided, even at the cost of more expensive additional hardware. Anything that is very difficult and time-consuming will probably not be done as often as it should be, and when it is done will probably be done wrong.

• If you need backup, you also need some high-quality training and support to the user organization. In particular, have a well-trained person responsible for restoring lost data from a backup. If any error is made and not detected until another backup is taken, recovery may be impossible.

• Always assume 2 generations of backup will exist and budget for the media to support both. Don't use 1 set of disks or 1 tape for all backup—rotate through at least 2.

A final issue in storage devices is the security of the media itself. Even floppy disks are expensive, and your company may find itself supplying many of its employees with them unless a form of security is maintained on the stock of unused diskettes. Once the data is actually on the disk, another security issue is raised—loss of company confidential data. If you have any data which is really confidential, make those who can gain access to it sign for it and take steps to prevent an unauthorized copy from being made.

Programs themselves may also be potential security problems. You are responsible for reasonable care to assure that programs which you purchase as a corporation and which are licensed to you for use are not copied illegally, expecially for wide distribution. The problem may be critical if you have a special program or the source code of a program which are protected by a non-disclosure agreement. These will always make the corporation liable for damages caused by the disclosure of, or distribution of, the information by employees.

PRINTERS, PLOTTERS, TERMINALS & COMMUNICATION

The connection between the microcomputer and the outside world is through a set of devices often called "peripherals" because they surround the computer and make it accessible to the human beings who must use it for the access or update of information. Peripherals can be either locally connected or attached via communication lines, normally the telephone system. They can also be either receive-only, permitting someone to see results but not to enter data (e.g., a printer), or "KSR," keyboard send and receive. Most KSR devices are terminals which consist of a CRT display and a keyboard while most RO devices are printers. The name "terminal" has become associated with CRT devices because they are the most common peripheral device.

TERMINALS are a key element in microcomputers because each system must have at least one for the communication with the operating system that allows program loading to be requested and which in general controls the actions of the system. Systems which support multiple users may have several terminals, and executive workstation systems designed for single-user operation may have the terminal and keyboard built into the computer itself.

The first user issue in terminal evaluation is the size of the display itself. There is a very real commercial standard—24 lines of 80 characters each. Any terminal which displays less information than this will probably be inconvenient operationally because terminal operators are conditioned to the larger display. In programming or spreadsheet development, a small display is an inconvenience which can slow the work and impare the visualization of the entire process. In word processing, a small display may prevent showing the information as it will appear on the printed page. If a display is larger than the standard, it may be difficult to find programs which will use the extra space to any advantage, but the additional display area will certainly not be harmful. Some systems will display up to 132 characters per line, useful if you want to display on a terminal what would normally be sent to a computer printout. Others will allow a special 25th line called a "status line" where the programs can display a message to the operator without inserting it into the middle of the work being done. Screen size is also important; for extended use, the screen should be at least a 12-inch diagonal. DON'T BUY A SYSTEM WITH A LIMITED DISPLAY CAPABILITY OR WITH A VERY SMALL SCREEN. If you do, a large number of target users will have to go to another system for reasons of program demands or readability, and your standardization effort is doomed from the start.

The question of integral terminal versus separate terminal has been much-debated. Systems which have a built-in terminal use a video monitor for display, and the display information is held in actual memory within the computer. This memory may use up some of the address space of the computer (an 80 x 24 screen is 1920 characters), but on the more modern 16-bit systems, the loss is not even observable. The primary advantage of "memory-mapped" systems is their speed of display and flexibility of formatting. A memory mapped display will often be able to "paint" an entire screen of data in far less than half a second and will be able to control each point on the screen for graphic display. The smallest point of screen which can be addressed to "paint" it a color or illuminate it as a point. Graphic display systems often describe their capabilities in terms of "resolution" and provide the number of horizontal and vertical pixel

Microcomputer Buyers Guide
Personal/Microcomputers Used In A Corporate Environment

coordinates which are valid. The Hewlett-Packard 2623A terminal, for example, advertises a 512x390 pixel resolution, similar to that of most popular color-graphics microcomputers.

Freestanding terminals are connected to the computer system through a "serial communication port" and cannot receive data to display fast enough to display a screen in less than 1 second, with 2 seconds the most common value. More significantly, separate terminals will normally not provide "point graphics" capability unless they are specially designed for graphics work (and therefore more costly).

Probably the most significant problem with the separate terminal is one of COMPATIBILITY. Terminal display requires not only that characters to be displayed be sent to the terminal, but that special formatting commands be sent for such actions as clearing the screen, moving the cursor, and controlling the intensity, normal/reverse character format, and blinking of the display. The terminal manufacturers have each designed unique control codes for their systems, and microcomputer manufacturers cannot control which terminal type might be attached. Users of microcomputers with integral terminals such as the Apple and IBM PC may purchase a software package which performs complex display management and it will run properly without custom adjustment, but owners of separate-terminal systems must normally perform an "installation" where they provide a special program with the identity of their terminal and which then sets the proper program steps to use that terminal properly. Installation procedures are not difficult, as long as the terminal you are using is one of those which the writers of the software envisioned as being supported on the system. SOME TERMINALS WILL NOT WORK WITH ANY POPULAR SOFTWARE PACKAGE WITHOUT CUSTOM PATCHING, A TASK FOR A FAIRLY SOPHISTICATED USER, OR WITHOUT ACTUAL MODIFICATION OF THE PROGRAMS, A TASK FOR A SPECIALIST. When you buy a terminal for a microcomputer, get one in common use or one which is compatible with a common type. The most popular microcomputer terminals are those costing less than $1,000, such as the Lear ADM-3, the Hazeltine 1400, the ADDS Viewpoint, or the Beehive. DEC terminals such as the VT-52 and VT-100 are also widely supported. But if you have specific software in mind for the system, be sure that your terminal type is supported by it before purchase.

HARD-COPY devices are becoming more common on corporate microcomputers as the device costs fall. Having a printer on the system is a significant advantage for nearly all types of users, and at least the ABILITY to support a popular printer should be considered mandatory in corporate microcomputers. Printers may be either dot-matrix or fully-formed character in their mode of creating the character impression. If the output of the system must approach typewriter quality, it may be necessary to go to a fully-formed character printer, usually called a "word processing printer," but other users will find the dot-matrix systems satisfactory. These printers form letters and figures by combining a closely spaced series of dots. The number of dots and the dot spacing control the quality of the image, and such printers are often advertised as "7 by 9 matrix," meaning that there are 7 possible dot positions across the character horizontally and 9 vertical rows of these 7. This 7x9 structure is probably the minimum for proper readability, but it is best to look at print samples to be sure the quality is adequate. Some dot systems will print multiple passes on the same line, offsetting the subsequent passes slightly to overlap the dots. Very good printers of this class can produce a character which appears to have been typed.

Here are some tips for printer purchase:

• Be sure that the print quality is adequate for your use. Print samples are always easily obtained.

• Printer speeds vary considerably, but dot-matrix printers are almost always faster than the letter-quality units. A speed of 120 characters per second is minimum if long documents are to be printed; less will tie the system up so long it will interfere with the productivity of the user. If you have a few documents which require high quality but most need printing speed more, consider one of the multipass dot-matrix units or get several letter quality printers and share them among all users.

• If a printer is to be used constantly, expect to pay at least twice the commonly advertised price for a unit of that speed and get a commercial-quality unit. Such units will advertise a DUTY CYCLE, the percentage of time which the printer is designed to be in use, of at least 60%. Also pay attention to the ribbon life on such units; changing ribbons several times a day may be difficult.

• Be sure the printer mates with the computer. Most printers are attached via what is called a "Centronics-compatible" or "parallel" interface, but some use a serial interface of the same type used to attach a terminal. Most printers are designed for parallel attachment, so if you have a computer which has only serial capabilities, expect to pay more. Word processing printers may also have special control codes for microspace justification on print lines, superscript or subscript, or other special printer features. These codes, like CRT format codes, are often incompatible among vendors. If you want a letter-quality printer on the system, find out which is supported before buying.

• ALWAYS BUY THE PRINTER CABLE TO ATTACH TO THE COMPUTER FROM THE COMPUTER VENDOR. The pin assignments on such cables are never standardized, and hours of frustration can result from trying to rewire an "inexpensive" cable purchased elsewhere. Your high-quality engineers, managers, and executives can probably figure the problem out in time, but is a savings of a few dollars worth it?

• Try to see the PRINTER you want running on the COMPUTER you want. There may be special setup needed for it, and you can get that information at the demonstration. If possible, have the printer dealer PRESET THE PRINTERS DELIVERED TO YOUR COMPANY FOR THE TYPE OF SYSTEM YOU WILL USE.

• Don't make concessions or pay for extended warranties or special maintenance on most printers. If you are going to buy a lot of them, buy in quantity with some spares and just return units which have failed for factory repair.

PLOTTERS are another hard-copy device gaining in popularity. A plotter may be sold as a combination plotter-printer or may be a specialized and separate device. Plotters MUST be used in conjunction with plotting software packages or custom programs, so it is VITAL that you research the type of plotter the software will require and purchase a unit which is compatible. You will almost certainly not want to give every computer user a plotter, so consider having a "pool" which can be drawn from when a need arises.

COMMUNICATION is one of the fastest-growing areas of microcomputer interest. It refers to the ability of the computer to attach to peripheral devices or other computers via telephone channels or other means over relatively great distances. Communication requires software support either directly from the operating system or through a purchased software package, but a minimum hardware support is also required. All data communication connections to microcomputers are through serial communication ports sometimes called "RS-232 ports" after the standard of the Electronic Industries Association standard which describes them. Simple micro-to-micro communication can be supported by almost any computer which has such a port, but attachment to a larger system is the goal of many corporate users, and that may require special hardware:

• Attaching to most IBM systems is best using one of IBM's SYNCHRONOUS protocols. The serial port on nearly all microcomputers is ASYNCHRONOUS ONLY. If you need a synchronous connection, such as one which emulates IBM's 3270 or 3780 devices, you will need a synchronous communication port. If you will be operating with IBM SNA, you'll need SNA software for your microcomputer and a serial port which is described as SDLC/HDLC, or BIT SYNCHRONOUS.

• Some communication applications require that the microcomputer answer and hang up a "data call." This requires control of more of the RS-232 interface than most systems allow. If you must operate your application in AUTOMATIC ANSWER MODE, or if you employ a HALF-DUPLEX transmission method for the connection, you will need a serial port which has FULL HANDSHAKING on the RS-232 interface.

• Communication with other computers require either a direct cable connection or a phone connection supported by communication equipment called "modems." Be sure that you get

Microcomputer Buyers Guide
Personal/Microcomputers Used In A Corporate Environment

support from a supplier experienced in data communication for the proper equipment. MOST CORPORATIONS SHOULD CONSULT A DATA COMMUNICATION EQUIPMENT DISTRIBUTOR FOR THESE NEEDS RATHER THAN A RETAILER, because of the possibility of more complex communication needs.

The questions of source of supply and point of service apply more frequently to peripheral devices in general than to the remainder of the system components. While many vendors will supply printers, plotters, and terminals for the microcomputers they manufacture, most buy such devices on OEM contract from peripheral manufacturers and apply an additional markup. Often the matching peripheral with the same logo as the computer will cost 30% to 50% more than the same device from its own manufacturer's stock. The problems with direct purchase of peripherals are the identification of the original or compatible manufacturer and the detection of "value added features."

Most microcomputer vendors will not readily identify the source of their peripherals, so your best source of that information is probably the inspection of the vendor's device and comparison of it to other similar products. Sometimes a retailer will know which unit is being resold, and sometimes the manufacturer of the printer or terminal itself will admit to making the product for the computer manufacturer to resell. Once you feel you know which unit is being remarketed, try to get a demonstration of the source product on the microcomputer system. If it works in all ways, you can substitute the less expensive product.

Even if you positively identify the original manufacturer of a peripheral and locate the source model from that original vendor, the substitute may not operate properly because of a change made to the unit by the microcomputer manufacturer. The change may be anything from a simple change of the cable wiring to actual circuit or logic changes. DON'T TRY TO REWIRE UNITS TO SAVE MONEY. If the substitute is not functionally equivalent to the original, don't buy it.

SERVICE of peripheral devices at low levels of use is probably best handled in the way recommended by the vendor; return to the place of purchase or through a national maintenance company. Corporate buyers who expect to use large quantities of a device should consider buying spares and returning failed units to the factory for refurbishment. Very inexpensive printers may actually be used for several years and discarded, but questions of depreciation may arise and should be checked with the financial departments involved.

THE MICROCOMPUTER ITSELF—CHIPS & BUSSES

Nearly all personal and small business users and most corporations will find that the evaluation of software, storage, and peripheral features against the profile of corporate needs will nearly select a vendor without regard for the type of microprocessor or technical features of the computer itself. Some users with very large purchase plans or with strategies for the long-term employment of microcomputers may wish to evaluate the technical features of the CPU in order to project what MAY become available for that system in the future. Users who contemplate the development of their own software will certainly want to review CPU features.

The CPU itself is made up of the microprocessor which actually executes the instructions written by the programmers and a standard interface to other hardware elements called a "bus." The other hardware components such as the disk controller, the peripheral controllers, and the memory of the system attach to this bus for communication with the microprocessor. Most of the technical features of the CPU derive from the characteristics of the microprocessor chip or of the bus.

Microprocessors are typically a single semiconductor "chip" which contains the logic necessary to interpret instructions and perform tasks. The tasks which a microprocessor can perform and the format of the command which causes it to perform a task differ from processor to processor. The "bit" identification is a measure of the number of binary digits the chip can move to or from memory in a single access. Since 8 bits make up a single character, an 8-bit system can handle 1 character at a time while a 16-bit can handle 2. The newest 32-bit processors, not yet in common use in commercial products, could handle 4. Unfortunately, the issue of 8, 16, or 32 bits has become more an advertising one than a technical one because of multiple definitions of what the terms mean and controversy over whether the issue has any real meaning. The Texas Instruments TI 99/4A, a popular home and game system, has a 16-bit microprocessor chip (the TI 9900) while the Xerox 820, a business computer, uses the 8-bit Zilog Z-80.

A more valuable way to evaluate microprocessors is by individually defining their characteristics. Here is a list of the most popular microprocessor chips, their strengths and weaknesses, and some systems which use them:

- The 6502. This is one of the oldest of microprocessor chips and is technically an 8-bit unit. The 6502 has a restrictive set of instructions, but it is FAST at what it can do. In special applications, the 6502 will out-perform all but the fastest and most modern 16-bit processors, but it is difficult to program in machine language, and the microprocessor features which make high-level languages like BASIC or Pascal efficient are lacking on the 6502. As a result, commercial 6502 systems do not perform well in benchmark tests where such languages are used. There is no accepted "standard" operating system for the 6502, so there is little vendor-to-vendor compatibility with its microcomputers. The chip is used in the Apple II, ATARI, and Commodore computers.

- The Z-80. This is the most popular microprocessor chip ever produced. The Z-80 was designed to be compatible with 2 of Intel's early and popular microprocessors, the 8080 and 8085, and machine code written for the Intel chips will directly run on the Z-80. All of these chips offer improved features for high-level languages, but the Z-80 provided extensions to improve performance even further. Digital Research CP/M is popularly used on the Z-80 computers, and the combination of CP/M and Z-80 is probably the most commercially popular today. A Z-80 with the PL/1 language was the champion of the benchmark tests for 8-bit computers in a recent magazine survey. Cromemco, North Star, Intersystems, the Xerox 820, and the TRS-80 models I, II, and III are examples of Z-80 microcomputer systems, and larger vendors such as Wang, DEC, and Datapoint use it in some of their products as well.

- 8088. This microprocessor was made popular by IBM in its personal computer. The 8088 is a modification of a 16-bit processor called the 8086, with which it retains full instruction compatibility. The change, producing a chip with the instruction power of the 8086 but with the ability to use the less expensive 8-bit memories. The 8088 is not compatible at machine code level with the earlier Intel chips, but a program to translate the assembly languages is available. The instruction set of the 8088 is very rich and can support high-level languages well, but the chip is fairly new and extensive programming support is just now becoming available. The dominant 8088 operating system is Microsoft's MS-DOS, used on the IBM PC as PC-DOS. 8088's are used in the IBM PC and its imitators, as well as in the DEC Rainbow and TI Professional computers.

- 8086, 80186. These are Intel's 16-bit microprocessors, capable of high speeds, efficient memory use, and good support of high-level languages. Microsoft's MS-DOS and CP/M-86 will both support these chips, and many commercial systems are being built using them. 80186 has a slightly expanded instruction set, but is downward-compatible with the 8086 and 8088. The 8086 family is used on the Burroughs B 20 computer, the Intel commercial computer systems iTPS and iDBS, the Convergent Technologies computer, the IBM Displaywriter, and the Wang Professional computer.

- 68000. This chip is the contender with the 8086 for the top new microprocessor. The 68000 is sometimes called a 32-bit system because its internal operation can handle 32-bit values, but in its most common form it is actually a 16-bit system. The 68000 probably has the best instruction set for high-level languages, and its similarity to the instructions used by the popular DEC PDP-11 minicomputers has contributed to its popularity with users. The 68000 will support the largest address space of any popular microprocessor, and its instruction set is ideal for scientific calculations. This combination makes the 68000 a superior scientific performer, second only to systems which have special

Microcomputer Buyers Guide
Personal/Microcomputers Used In A Corporate Environment

arithmetic or array processors installed. A disadvantage of the 68000 is the fact that the dominant operating system at the moment is UNIX, a system for which commercial business software is less common than for others such as CP/M. The 68000 is used on the Cromemco, Fortune, NCR Tower, and IBM professional/scientific computer.

Other microprocessor chips such as the Motorola 6800 and 6809, the Texas Instruments 9900 and 9995, and the Zilog Z-8000 may be found in some business systems, particularly in those manufactured by the company which developed the chip. The software base for these systems is limited by the fact that their installed base is too small to attract significant third-party developers. Corporate purchasers should explore their intended use carefully before selecting any microprocessor which has limited software support.

Processor designations such as those shown above often have a letter suffix associated with them, e.g., "Z-80A." This suffix indicates the maximum clock speed of the chip a measure of the relative speed of its operation. Early products typically have a lower clock speed, and the speed increases as technology improves. The basic Z-80 has a 2 million cycle-per-second clock while the newer "B" version will operate at 6 million cycles per second. Clock speed, however, has less impact on system performance than many other attributes such as the features which support high-level languages. The only real measure of system performance for users whose applications demand speed is a benchmark test of the systems being considered.

MULTIPLE MICROPROCESSORS AND CO-PROCESSORS are becoming more common as users demand more performance from small computers. Systems which employ multiple microprocessor chips may do so to provide the speed and other advantages of 16-bit systems while retaining compatibility with the older 8-bit software. DEC's Rainbow system and the Cromemco CS3-HD5-E are examples of such computers, giving CP/M compatibility in 8-bit mode and 16-bit performance where 16-bit software is available. The former uses a combination of the Z-80 and the 8088, and the latter uses the Z-80 and the 68000.

Another use of multiple microprocessors is to absorb the processing needed to control input-output devices such as disks, communication lines, or terminals. This allows the "real" microprocessor to spend more of its time executing programs, thus improving performance. A multiprocessor system where a second microprocessor is used to control the peripheral devices and disks is sometimes called a "service processor" system because the second processor performs all I/O services for the main processor. Service processors may be conventional microprocessors or may be specially-designed for the application.

One of the fastest-growing multiple processor applications is that of the scientific or arithmetic co-processor. This is a special-purpose computer chip designed to operate in conjunction with one of the business microprocessor chips already discussed. The best example of this concept is the Intel 8087 High Speed Arithmetic Processor. Microcomputers which employ the 8086/8087 pair are incredibly fast in the execution of certain mathematical operations and especially in scientific applications. A major corporation recently tested an 8086/8087 combination running a FORTRAN test program against other microcomputers and found that the execution on the 8086/8087 was so fast it could not be timed! Co-processors WILL NOT AFFECT ALL APPLICATIONS, and for them to operate at all THEY MUST BE SUPPORTED BY THE APPLICATION PACKAGE OR PROGRAMMING LANGUAGE. If you intend to select a system because a co-processor is available for it, be sure that the system's programs will use it. Otherwise the chip will do no more than help heat the computer area.

BUSSES are almost an external extension of the microprocessor, the way in which it extends its control to the rest of the system. Microprocessor busses are normally defined by a manufacturer and may be proprietary, but many have become so widely used that they have become de facto standards and are accepted by other vendors as well. Some are even formally recognized and standardized by a standards organization.

The importance of a bus for the corporate buyer is in this issue of standards. If a computer with a standard bus architecture is purchased, memory, controllers, and other integral parts of the system can be expected to be produced by many manufacturers, resulting in a wide variety of products and competitive pressures to lower costs. There are 2 popular "standard" busses widely used on microcomputer systems:

- S-100. This is a very old standard developed for the Intel 8080 and later popularized for the Z-80 microcomputer. The IEEE recently adopted a formal standard for the older bus called IEEE-696, but many systems and many components are already marketed which do not conform to the standard. The original S-100 bus was an 8-bit system bus, but the IEEE-696 will support 16-bit microprocessors as well. Buyers of 8-bit systems can expect some real benefits in purchasing a system using this bus because there is a significant amount of special hardware available for its. Teamed with a Z-80 processor and the CP/M operating system, the S-100 probably makes up the most flexible 8-bit microcomputer system.

- Multibus. Intel designed the Multibus for its development systems, but the high quality and flexibility of the bus has encouraged its migration to commercial products. The IEEE has adopted a standard, IEEE-796, for the Multibus. There is a growing support for the Multibus to become the standard for commercial 8086 systems, particularly for those which are designed for wide commercial and industrial applications. Burroughs, NCR, and other vendors supply some computer systems with Multibus compatibility.

Other bus structures exist, such as the Versabus, VME bus, STD bus, Eurocard, etc. While any standard or common bus architecture can be a benefit to the user, it is doubtful that the use of any bus other than the S-100 or Multibus will offer significant benefits to corporate buyers who are not selecting systems for some specific scientific or process control application.

Most corporate buyers will probably not care about either bus structure or microprocessor type. Here are some ways to see if you are a buyer who might:

- If some of the software, operating system, or language issues discussed restrict you to particular packages, you must be sure that the type of microprocessor and bus structure of the system support those packages.

- If you must interface with a specific piece of hardware such as a scientific instrument or magnetic tape unit, you will have a higher probability of finding a suitable device controller if the bus architecture of the computer selected is a popular standard.

- If you have a scientific, mathematical, or technical application for your microcomputers, it may be necessary to select a chip with high performance in hardware multiplication and division and/or with a mathematics co-processor.

The final issues in the selection of a corporate microcomputer based on the CPU are those of purchase point and service.

Most small computers are purchased from retail outlets, and a corporation may be able to get quantity discounts from such stores. The advantage of dealing with a retail store rather than with the manufacturer directly is the availability of third-party software and hardware for the system from the same source as the computer. IBM personal computers, for example, can be purchased from IBM or from one of many authorized dealers, but the dealers will also sell non-IBM software and peripherals while IBM will not.

Some manufacturers will execute volume purchase agreements with corporations who wish to acquire a large number of systems, and the discount can be significant. Others will permit corporations to buy under what is called an "OEM Contract," one which is normally executed by firms who will add software or other value to the computers and resell them. Beware of OEM sales; they often do not include any form of support or training and may even involve a more restrictive warranty.

Ideally, a corporation would like to have the same on-site maintenance available for its microcomputers as is available on the data center equipment, but that goal is difficult to attain. Only the computers designed for business and commercial applications are available with manufacturer maintenance

Microcomputer Buyers Guide
Personal/Microcomputers Used In A Corporate Environment

on-site; the others will rely on a national service organization for such capabilities. Unless the microcomputer you select is very common in your geographic area, you will probably find that maintenance organizations will be unable to fix problems as rapidly as you would like. Local computer dealers may be able to supply on-site maintenance also, but check with other users to be sure that they are actually qualified to do so.

Most corporations which purchase 20 or more systems would probably be best served by a form of "self-insurance"—stocking an extra system as a spare and substituting it for a failed system, returning the faulty unit to the manufacturer for factory repair.

MULTIPLE USER COMPUTERS

Most people, including corporate users, think of microcomputers in terms of the single-user system. While it is true that most microcomputers are used in that mode, there are a significant number of systems available which are designed for multiple users.

Most computers used in corporate data centers are designed to support multiple users, and it may seem attractive to extend that concept to corporate microcomputers. Multiple use in large computers is based on the fact that the central resource of the computer and its associated devices is both high-performance and expensive. It is thus desirable to share it among users, any one of which could not utilize enough of it to justify the cost. With microcomputers, the cost of the microprocessor chip is likely to be less than $20, and the cost of the user's memory less than $1,000. Slicing up resources like this may not provide the benefits needed to justify the risks or multiuser systems; reliance on a single system for multiple organizations, problems with data security, and performance problems due to over-utilization.

The simplest form of multiuser microcomputer systems is the multitask system which provides each user their own memory area but shares all other system resources among all users. This system is structurally similar to the larger multiuser mainframes, but is very susceptible to overloading because the inexpensive microprocessor chip is also shared. The next multiuser system is sometimes called a multiprocessor system or service processor system. In it, each user has dedicated memory and microprocessor resources and shares the files and other peripheral devices through a service processor.

Cluster systems are essentially multiprocessor systems but with the addition of support for "private" disk and peripheral resources and the ability to connect the users to the service processor via some form of communication connection. In practice, cluster systems can operate with a station separation of as much as several thousand feet.

Interviews with users have shown that most multitask microcomputers produce noticeable performance degradation when 3 or more users are sharing the system, and that many users find performance unacceptable with over 5 users. The multiprocessor systems with a single, shared set of disk devices for storage will normally support up to 8 users without problems, and cluster systems with properly allocated private resources will support over 30 users. Multitask systems will probably have a lower per-user cost than a single user system with similar features, and multiprocessor systems may have a somewhat lower cost as well. Cluster systems are actually more expensive than many single-user computers and thus must be cost-justified by the benefits of being able to share files among users.

Most corporations should not consider the use of multitask systems because their potential for degradation under load is too high. The other types of multiuser systems should be considered not for their potential cost savings but for their ability to permit easy exchange of information between users. Many such systems support a form of electronic mail, for example, and nearly all will allow users to share access to a large file so long as none are updating it. A multiuser system such as this can eliminate some of the problems with data dispersal which plague microcomputer applications in a corporation. Cluster systems in particular are useful in this mode because each cluster station is a complete computer, but each station can also access a single shared database through a service processor. That service processor can also support mainframe communication protocols which will give cluster workstations access to the databases of the data center and permit the data on the cluster to be loaded to the main computer system.

LOSS OF CONTROL is the biggest problem with multiuser systems in a corporate application. Such a system is potentially a small data center, operating in competition with the corporate facility and controlled not by a specialist but by a user organization. Other organizations may want to "tie in" to the new system, aggrevating the problem. For this reason, use of multiuser systems must be controlled carefully through the application of strict selection guidelines such as these:

- Don't consider multiuser systems except where the sharing of information between microcomputers or with the mainframe is a key issue in the application.
- Require that any shared-use system have a mainframe communication link and use that connection to extend at least an element of central data center control into the applications.
- Require that any use of such a system outside the original department be approved by a data processing review committee.
- Require that operators of such systems attend data center operations meetings and that they conduct similar status and information meetings with their users.
- Make it CLEAR that formal procedures such as these will accompany any installation of a multiuser system. An organization unwilling to undertake them should not be acquiring something as close to a corporate resource as a shared-use computer system usually proves to be.

ALTERNATIVES TO MICROCOMPUTERS IN A CORPORATION

Microcomputers selected by user organizations to perform local tasks are only one way of getting such tasks accomplished, and often not the best way. Any use proposal to purchase microcomputers should be subject to a rigorous justification—WHY CAN'T THE JOB BE DONE WITH THE DATA CENTER COMPUTERS?

In many corporations, the growth of microcomputers is not the result of a legitimate and unique need for distributed computing power but a result of frustration and animosity with the corporate data center. This is a wholly unsatisfactory motive for purchasing additional computers and should never be accepted by corporate management. Here are some things which may allow the data center to absorb some or all of the tasks proposed for microcomputer systems:

- Special application packages. More and more mainframe vendors are making spreadsheet, word processing, or data handling and reporting packages available. Often these packages are superior in features to their smaller counterparts, and the data collected resides in the corporate data center where it can be used by the company at large for management analysis and reporting.
- Low-cost terminals. Newer terminal devices are becoming very inexpensive, making the extension of the power of the data center to the user more practical. Explore the new offerings of your vendor, and check to see if compatible devices are being offered elsewhere. There is a new wave of devices which allow a simple, low-cost CRT to emulate an IBM 3270-type display station. The use of this device can extend access to IBM mainframes to users of minicomputers whose terminals are incompatible with the IBM systems. They can also allow a corporate user to purchase new terminals for $600 or less and attach to their IBM hosts.
- Programmer aids. Many times the problems with the use of the data center are centered on the time and cost associated with the development of a new application. Newer programming productivity aids can reduce this time and cost, making it possible for corporate programmers to provide customized applications to user organizations at less than the cost of acquiring microcomputers and programs for them.

The corporate data center is always the prime alternative to microcomputer use, but there are other options available:

- Timesharing and service bureau use. When a task is

Microcomputer Buyers Guide
Personal/Microcomputers Used In A Corporate Environment

recognized to be of short duration and cannot be supported on the data center, going to a service bureau or using a timesharing service is an alternative. It's an expensive alternative, though, and the risks of using it in terms of loss of application control and cost control are probably even greater than with microcomputers. If a timesharing/service bureau contract is contemplated, EXPECT TO REVIEW ITS STATUS FREQUENTLY TO BE SURE THAT IT DOES NOT CONTINUE TO A POINT OF EXCESS. Three weeks on a timesharing system may be a reasonable alternative to corporate microcomputer usage but a year is probably not.

- Renting time from other users can be considered if the application is non-critical or if your corporation and the other user have a contract covering mutual renting of capacity which would guarantee the time you need. Otherwise it is an option with all the problems of a service bureau plus the uncertainty of whether the time will actually be available.

With any use of external resources the issue of application program development or availability must be addressed. If you buy time from your local timesharing house, will it have the program you need to run? How much will that program cost, if anything, in addition to the basic service? If you develop a program for that system, will it be of any use to you in the future? In general, going outside for service will not be viable alternative. Explore it to be sure, but be cautious.

The final alternative to microcomputers is perhaps one of the best—no computer. Manual operations are frequently NOT subject to automation, trade press articles notwithstanding. This is particularly true where the level of service offered to customers or suppliers through the task being evaluated is very high and where the variability of the operation from day to day and from one client or account to another is significant. Many executives and managers and nearly all ambitious professionals would like to have access to a company computer to increase their computer literacy or to show off to visitors. If your corporation feels that the benefits to be gained as an organization offset the cost, the provision of computers for what is almost personal use may be justified. If so, be sure to FREELY ADMIT TO THE STRATEGY rather than cloak it in unnecessary "target applications" and "justified uses." If your make a department invent an application to justify a system, that application may actually be attempted.

Where applications are actually being screened, look at the possibility of manual solutions as well as of data center solutions. The issue of whether to require a manual alternative be considered first is one relating to the corporate information strategy. If you desire to have all corporate operating data on a computer file for management information purposes, accept a data center solution which is cost-effective even if the manual alternative is more so. With microcomputer applications, remember that having data on a floppy disk for a microcomputer does not make it available to the corporate MIS system. If you want integrated data, use the data center or use communication to integrate the microcomputers with the data center.

CORPORATE PURCHASE OF MICROCOMPUTERS—A FINAL LOOK

The use of microcomputers in a corporation must be reviewed carefully because of the risks of loss of control of the systems. Such a loss could cause the installation of many incompatible systems, reduces the ability of the corporation to collect and digest information at a central point, reduce the quality of service to clients, suppliers, and management, and cost nearly as much as a major mainframe computer. Corporate purchase should be controlled by a central point, either in the purchasing organization (who at least are likely to find out that microcomputer purchases are being contemplated) or in the data processing organization.

The corporate buyer must first establish whether the way in which the corporation does business is suitable for the employment of microcomputers, then evaluate the technical features of the systems available in the light of possible corporate applications. During this evaluation, flexibility must be a key issue because the wide range of business activities present in today's corporation creates a wide range of demands which may be difficult for some systems to meet. Purchase of a limited system can only lead to the need to either discard it at a loss or to purchase systems of another type for other applications, decreasing the compatibility between microcomputer users and increasing risks.

Features in a corporate selection must be judged against GENERAL POLICIES AND REQUIREMENTS rather than against either specific applications or prevailing programmer attitudes. Programmers should not select corporate computers because they lack an understanding of operating department issues and because they tend to emphasize esoteric features which the average user could not possibly employ properly. Department managers are likewise blind to the technical issues of flexibility of design and proper support for future growth. A committee for evaluation is probably the best answer, but BUSINESS ISSUES SHOULD BE PARAMOUNT and technical input should only direct the selection process within the boundaries of operational suitability.

Service and maintenance policies should be evaluated by someone experienced in such areas; the data center operating management personnel are normally a good choice. While it is unreasonable to expect large-system maintenance from a computer costing only a few thousand dollars, many users will be conditioned to the idea that "if it breaks, someone will come to fix it" and must be specifically disabused of this thought if that is not the case. The idea of self-stocking of spares and factory return for service should always be explored where there are a large number of systems involved. On-site maintenance is not normally needed for small computers; if the application is that critical, it should probably be done on data center equipment in any case. Maintenance quotes from the dealer, the manufacturer, and third-party firms qualified and authorized to maintain the systems should be requested. It is also a good idea to request maintenance references or to attend a meeting of the local user group for the computer system to get maintenance experiences. You should expect to poll the attendees rather than rely on spontaneous comment; happy people will frequently not speak up.

The issue of whether to undertake a formal purchase procedure is always a difficult one. Requests for Proposal are a corporate way of life, but may be totally new to some suppliers of small systems. When you are selling a unit with an average profit of less than a thousand dollars, you may be less than enthusiastic about taking several man-weeks to review and respond to a proposal. Try doing a study with your own corporate resources rather than asking for formal proposals. In that way, vendors with lower prices (and therefore lower margins) will not be discouraged from responding. If you do elect a formal bid process, try to be as explicit as possible in what you want from each bidder rather than to ask for free system analysis. Again, the microcomputer systems do not support the massive proposal efforts you may have seen from mainframe vendors. A corporation recently sent a 350-page RFP to 13 microcomputer manufacturers and received no responses—the gross profit on the sale would have been only about $10,000, and no vendor could justify the effort.

Many corporations will want to look carefully at the issue of financing. There are few general tips available in guiding a major business through the purchase-versus-lease decisions; the best course will be determined by cash flow considerations and tax considerations. Ask your financial department to prepare a comprehensive study of the alternatives such as a discounted cash flow analysis, and take the most favorable option.

Another difficult issue is that of contract terms, damages, and non-performance. There are a wide variety of legal remedies available to corporations who have been damaged due to fraudulent misrepresentation or negligence, but they may take years to apply and often require embarrassing admissions in court. THERE IS NO SUBSTITUTE FOR KNOWING WHAT YOU ARE DOING, legal remedy or otherwise. Rely on the vendor only for basic questions of features, capacities, and so forth. You must make your own judgements on suitability. This is particularly true in dealing with local retail suppliers. The courts are going to be reluctant to believe that XYZ Corporation, one of the top thousand in the U.S., was mislead and damaged by improper technical representations of a 1600-square-foot computer store.

Warranties, particularly of the "return it in 30 days for full refund" type, are important to small users but less so to corporations. You should not buy something if you are not sure it will work, and

Microcomputer Buyers Guide
Personal/Microcomputers Used In A Corporate Environment

there is a good chance that the first application of a system in a corporate environment will take so long to develop that the warranty will expire before it gets started. If you can get a "return" warranty on equipment, set up a study group to go to work on the system so that is is evaluated within the time frame. DON'T let the requestor run the group—there may be a reason to let the return period expire with the equipment untested! Warranties against manufacturing defects and failures should cover 90 days, the normal "infant mortality" period for electronic equipment. If the manufacturer normally warrants a shorter period, have it extended in the contract for sale OR GET AN EXTENSION IN WRITING. Don't inventory systems at your location for long periods of time, or the warranty will expire before the unit is uncrated. Either keep the gear at the vendor or dealer location until it is needed or have the warranty period begin with the setup of the system by the dealer or vendor.

Some prospective corporate purchasers may find that none of the guidelines for technical selection and screening seem to apply well to their proposed use. If you cannot find any issues specific enough to either justify a particular system or rule out a large set of alternatives, you may have to base your selection on the single issue of FLEXIBILITY. The following points summarize the characteristics of a professional workstation with the highest business and technical flexibility:

- Built by a manufacturer who also makes larger computers, providing an upward migration path and assuring the corporation that the vendor understands large-company issues.
- A 16-bit microprocessor and the capability to support at least 256K bytes of memory.
- Floppy disk storage of at least 500K bytes on 2 disk drives, with hard disk support optional.
- Graphics capability with a 512x390 resolution or better.
- Operating system which supports the popular langauges such as BASIC, COBOL, Pascal, etc.
- Terminal emulation support for a popular terminal such as the DEC VT-100 or the IBM 3270 series.
- Supporting a popular word processor and financial spreadsheet package. What's popular? If you can get a book of examples or operating procedures for the package at a well-stocked book store, it's popular.
- Supporting a data entry and file handling package, sometimes called a "database" program.

If you select such a system for your corporation, it is unlikely that you will encounter any proper microcomputer application which cannot be satisfactorily supported.

Microcomputers can be a powerful tool in improving the productivity of the key personnel in corporation, possibly the greatest opportunity to address this key area in history. To fulfill the potential of these small computers, a corporation must apply them with the realization that they are neither toys nor status symbols but tools which can be misused with disasterous consequences. If your corporation has already purchased microcomputers without controls, you are probably facing the problems with incompatibility and unhappy users already. In this mode, there is no solution short of a full technical review of the systems at hand and a recommendation of what can be done with unsatisfactory units. Sometimes a donation of computers to charitable institutions or schools will reduce the loss associated with moving away from a given system or systems. The cost of retrenching may be high, but the cost will be even higher if the uncontrolled growth of microcomputer use continues. You cannot control a multivendor environment which developed without central guidance. You can only start over.

Avoiding the agony of restructuring an existing microcomputer environment within a corporation can be avoided by anticipating the use of the systems. Nearly every corporation can expect to have some microcomputers installed for professional and managerial use within the next 3 years, and over half already have some systems in use. The opportunity to control their application is fading fast, and can be grasped only by cooperation between the purchasing, data processing, and user organizations:

- through purchasing review and flagging of microcomputer purchase requests,
- through technical studies to determine the most suitable microcomputers IN ADVANCE OF THE NEED TO INSTALL THEM,
- through department understanding that "modern" and short-term solutions to local processing needs cannot be permitted without attention to the overall business impact,
- and finally through executive determination that the application of microcomputers will be MANAGED rather than just allowed.

• END

IBM PC Compatibility

Understanding & Evaluating Claims of IBM PC Compatibility

■ INTRODUCTION

The explosion of systems in the microcomputer market is partly the result of IBM's entrance into that market with systems that proved to be very popular. The enormous success of IBM's "PC" product line has led to a phenomenon familiar to mainframe users—IBM-compatible competitors, in this case, the rise of IBM PC-compatible systems which claim various degrees of hardware and software compatibility with IBM's PC. The compatibility of PC "lookalikes" or "PC clones" as they are sometimes called, is a more complicated matter than compatibility at the mainframe level.

Mainframe users usually find they need to ask only two questions:

• Can I run all the IBM operating system software, supporting system software, and applications software without alteration?

• Can I attach all my existing peripherals and terminals to the PCM system, and expect to attach new offerings from IBM?

The answer to both these questions is usually a yes with only a few qualifications when the user is inquiring about a 3030-, 3080-, or 4300-compatible mainframe "PCM" system.

Furthermore, PCM (plug-compatible mainframe) vendors have developed a tradition of ongoing support; users expect them to implement changes that match IBM changes, so that existing PCM systems can be upgraded to the equivalent of the upgraded IBM system.

In the microcomputer marketplace, the potential PC-compatible user needs to ask far more searching questions in order to gain a reasonable assessment of the degree of PC compatibility a system possesses. Manufacturers themselves do not agree on the most basic terminology. One manufacturer may say, "My system runs MS-DOS and can read and write in IBM-compatible data format, but the system is not PC compatible." Another manufacturer may say, "My system is IBM compatible because it runs MS-DOS" and totally disregard the fact that data is read and written in a different format on different capacity diskettes.

The questions a potential microcomputer buyer must ask to get a good picture of a system's compatibility are numerous, partly because of the lack of consistent terminology among vendors, and partly because compatibility problems arise in areas unexpected by users of traditional mainframes. These questions are as follows:

1. Can I run both compiled BASIC and interpreter BASIC programs written for my IBM system on the PC-compatible system? Can I run PC-compatible BASIC programs on my IBM system? Conversely, I take my IBM BASIC software, read it into my PC-compatible system, and use it to write BASIC programs?

2. Can I take the version of MS-DOS that is supplied with the PC-compatible system, and run it on the IBM PC? All versions? Conversely, can I run IBM's PC-DOS on the PC-compatible system? All versions?

3. Are the central processors in the two systems compatible? Are they based on the same or related microprocessors?

4. Can I take an IBM board or a board-level option, like the color/graphics display adapter or the expansion box, and plug it into my PC-compatible system? Conversely, are there any alterations needed on the PC-compatible system in order to attach IBM-supplied options and peripherals?

5. Is the keyboard supplied with the compatible system identical to the keyboard supplied by IBM? If there are additional keys on the compatible system how do they affect the compatibility of programs or data that have been generated with/without the use of these keys?

6. Can I take IBM programs that use the PC all points addressable (APA) graphics features and run them on my compatible system? Conversely, can I run graphics programs supplied with my compatible system (if any), and run them on the PC?

7. Can I take a diskette recorded by the compatible system and read it into an IBM PC? If I read it into the system, will the running program be able to manipulate the data without change or conversion first? Conversely, can I take a diskette recorded by the IBM PC and read it into the compatible system? If I succeed in reading the data in can the running program use it directly, without additions, deletions, or alterations?

■ SOFTWARE COMPATIBILITY LEVELS

Unlike mainframe systems software, which has had a long upgrade history, and hence is oriented toward modularity and device independence, microcomputer operating software tends to be closely related to the CPU and I/O structure of a system, and tends not to be completely portable. It is impossible to talk about hardware compatibility features without discussion of software to some extent. Conversely, when buying a software program designed to run "under MicroSoft's MS-DOS," the user has to inquire under which MS-DOS—i.e., under which implementation of MS-DOS. IBM's PC-DOS is one implementation of MS-DOS among many.

IBM PC Compatibility

Furthermore, there are significant differences in capability between the three major upward-compatible software levels available to PC users; i.e., between systems controlled by (a) ROM BASIC (without an MS-DOS or any other operating system), (b) PC-DOS 1.1, and (c) PC-DOS (2.0 or 2.1). PC-DOS 2.1 is very similar to (and replaces) PC-DOS 2.0, but includes several fixes and extensions of enhancements introduced for 2.0, plus the ability to run on PCjr.

These three levels are largely upward compatible, which is to say, programs designed to run on a BASIC system can usually run on a DOS 1.1 or 2.1 system but not the reverse, and programs designed to run on DOS 1.1 systems can usually run under DOS 2.1 but not the reverse. Furthermore, DOS 1.1 cannot handle hard disks and consequently cannot run on the PC/XT; it also is not supported by PCjr. DOS 2.0 runs on PC 1 and PC/XT diskette-based or hard-disk-based models. It has been replaced by DOS 2.1 which runs on PCjr diskette-based systems as well as on PC 1 and PC/XT.

There are also other operating environments available for the PC, like CP/M-86, concurrent CP/M, and p-System, but the primary focus of IBM and its competition, has been the development of the PC-DOS versions of the MS-DOS environments.

□ BASIC Compatibility

Can I run both compiled BASIC and BASIC programs written for my IBM system on the PC-PCM system? Can I run PC-PCM BASIC programs on my IBM system? Can I take my IBM BASIC software, read it into my PC-PCM system, and use it to control BASIC program development?

One of the fundamental differences between the IBM PC product line and PC compatibles is the way the BASIC interpreter has been implemented. IBM stores the BASIC interpreter and enough system control logic in its proprietary ROM so that a cassette-based PC can operate without a more elaborate operating system. Only disk- or diskette-based systems require PC-DOS. For diskette-based and hard-disk-based systems, IBM has further added capabilities to BASIC in certain add-on packages. For example, IBM's Advanced Basic level is achieved by adding the BASICA software package to the "primitive" BASIC capabilities in ROM. This combined BASIC level is equivalent to MicroSoft's Gee Whiz BASIC (GW BASIC).

Since the ROM is proprietary, most PC-compatible vendors have taken the MicroSoft GW BASIC package and altered it to match the combined capabilities of IBM's ROM BASIC plus BASICA, i.e., Advanced BASIC. This means that the IBM BASICA program itself cannot run on the PC-compatible system because its ROM-based functions are missing. The compatible vendor usually supplies equivalent BASIC all in software.

It would be possible, but not easy or efficient, for the PC-compatible vendor to design ROM so that entry points for calls to the IBM BASIC interpreter are coded with traps that are related to the appropriate software sequence for executing the instruction. Alternatively, the PC-compatible vendor could put BASIC in ROM, preserve entry points at the same addresses as IBM, but change the implementation of the coding in such a way that it is not identical, yet serves the same function as the corresponding code sequence. This is not easy either, and probably legally dangerous. All currently tested PC-compatible vendors implement BASIC totally in software. As far as we know, no vendor, not even Compaq, used either of these methods, but has gone the all-software route for BASIC. Consequently if third-party software vendors directly addressed ROM BASIC code sequences, the software would not be compatible. The upshot is that even some highly compatible systems have problems with interpretative BASIC programs, particularly if those programs include graphics-related sequences. See "Compatibility Levels" later in this discussion.

□ PC-DOS Operating System Compatibility

Can I take the version of MS-DOS that is supplied with the compatible system, and run it on the IBM PC? All versions? Can I run IBM's PC-DOS on the compatible system? All versions?

As stated earlier, IBM's PC-DOS is a modified version of MicroSoft's MS-DOS. PC-DOS 1.1 corresponds to MS-DOS 1.25; PC-DOS 2.0 corresponds to MS-DOS 2.0; and IBM PC-DOS 2.1 corresponds to MS-DOS 2.05.

MS-DOS is a fairly transportable generic operating system designed to run on the Intel 8088/8086 microprocessors or their 80188/80186 counterparts. As it is implemented on a particular set of chips with certain memory ROM and I/O structures, it must necessarily adapt itself to the individual architecture. Incompatibilities are likely to arise between one MS-DOS system and another, but these can conceivably be circumvented through software and/or firmware adjustments, if the vendor wants to.

MS-DOS has its own Basic I/O Subsystem (BIOS), designed to be implemented in user ROMs, in a manner consistent with the vendor's developing architecture. A PC-compatible system must take this BIOS and implement it in such a way that it is not identical to IBM's proprietary ROM BIOS code, yet provides the same functionality and interfaces, in a manner totally transparent to the user hardware and software. The BIOS contains code for the operating system bootstrap; for system checkout and testing; for interrupt vectors allowing BIOS-level interfacing for I/O manipulation; and for the initialization of memory, interrupt vectors, scratchpad, I/O, and flag values. It is particularly important that ROM BIOS interrupt vectors are initialized in such a way that they point to the same functions as found on the PC.

As is the case with the more advanced BASIC interpreters for diskette- and disk-based systems, IBM uses the existing (cassette-level) ROM BIOS together with a tailored software BIOS to achieve the equivalent of the all software BIOS supplied with MS-DOS. A PC vendor can implement this in various combinations of ROM and RAM providing certain addressing, interrupt and memory constraints are respected. For instance, bootstrapping the operating system requires that a specific diskette sector (sector #1, 512 bytes, on track #0 of head #0) be read into a specific memory location using a specific ROM BIOS interrupt (#13 hexadecimal). In addition, it is wise to keep the overall size

IBM PC Compatibility

of the compatible version of PC-DOS BIOS smaller than 2K bytes, because this is the conventional size for the system tracks on the diskette. The set-up procedure for many applications software packages requires the user to copy the operating system onto a systems portion of the applications program diskette, in order to make the whole diskette bootable. If the compatible vendor does not stick to the 2K-byte convention, the operating system may overwrite the applications program.

Another of the essential problems of claiming software compatibility is that a PCM vendor is always shooting at a moving target. As the primary systems vendor improves on or expands the system capabilities, new operating system releases (or new BASIC releases) may be needed to fully exploit the expanded system. A program designed to run under the old operating system release may not run under the new one without modification. For example, IBM's PC-DOS 2.0 (and 2.1) operating system versions were developed with the introduction of the PC/XT and PCjr respectively; either one will run on the original PC ("PC 1"). When IBM announced PC-DOS 2.1, it stopped selling PC-DOS 2.0 altogether; presumably PC-DOS 1.1 will also eventually be phased out in favor of the more mature version, since 2.1 runs on all systems designed for DOS 1.1. Yet, certain programs developed for operation under PC-DOS 1.1 will not run without modification under PC-DOS 2.0 or 2.1; IBM published a list of these when it announced PC-DOS 2.0.

The questions revolving around issues of compatibility might not be as difficult to resolve if PC-compatible hardware and software vendors would be more precise as to the specific PC model(s) and specific operating system version(s) for which compatibility is claimed.

For information on some specific differences between 1.1 and 2.0, see the discussion on file handling improvements under DOS 2.0 in the discussion on disk/data compatibility, later.

■ HARDWARE COMPATIBILITY LEVELS

The PC-compatible marketplace necessarily lags behind the PC product line from 6 to 18 months, because of the time required to develop and manufacture a system. Consequently, the PC-compatible vendors primarily target the original PC 1 running under PC-DOS 1.1. Some of the capabilities provided by PC-compatible vendors have been incorporated by IBM into the PC-DOS 2.0/2.1 and PC/XT product levels. By the beginning of 1984, several PC vendors were announcing XT-compatible products but testing data is not yet widely available for evaluating this level of compatibility. Consequently, the following discussion is primarily focussed on systems competing with the PC 1.

□ CPU & Microprocessor Compatibility

Are the central processors in the two systems compatible? Are they based on the same or related microprocessors?

The "central processor" in a microcomputer system is usually equated with the primary microprocessor used to execute the primary instruction stream. The term CPU is hardly used at all. For the IBM PC, the primary microprocessor is an Intel 8088, which is part of a compatible family that includes 8085, 8086, 80188, and 80186 8-bit and 16-bit chips as well as upward-compatible 80286 32-bit versions.

Both the 8088 and 8086 use the same instruction set and addressing based on a 16-bit word. The 8088 uses an 8-bit external bus so that full 16-bit transfers require 2 steps. The 80188 is an enhanced version of the 8088, still retaining an 8-bit bus, but with a higher level of integration, faster clock, more on-chip functions and an external instruction set. The 8086 is the same as an 8088 but with a 16-bit external bus allowing full word transfers each read or write cycle. The 80186 includes the 16-bit bus, plus enhancements similar to those of the 80188. Specifically, chip functions such as DMA, interrupt control, timers, wait-state logic, and chip select are right on the chip.

It is conceivable that a fully PC-compatible microcomputer system could be designed with any of these chips, by firmware or software programming around differences in implementation of essential functions. The new 80188 is specifically designed as an upgrade to the 8088; the 8086's difference in data bus width is compensated for by almost identical interfacing to the bus. Nevertheless, all currently delivered systems that claim full PC-compatibility (hardware, code, disk, keyboard, I/O, software) use the 8088 as a base. It seems logical that a company primarily interested in marketing a highly compatible system would choose the 8088, which is identical to IBM's microprocessors, and thus cut back on the compatibility adaptations needed. On the other hand, if a company wants to implement an MS-DOS system with disk/data compatibility, but with enhanced features over what IBM offers, the logical choice is the more powerful 8086 and 80186 processors.

The compatibility of the basic microprocessor component is only the first issue, however. When a large systems user asks whether two processors are compatible, and receives an affirmative answer, a piece of information that is of primary importance has been given. If the two processors are identical, they must be largely, if not completely, compatible.

One of the differences in viewpoint lies in the amount of logic assumed to be included in the "central processor." If a user says a system is based on an Intel 8088, he does not usually mention that the interrupt control logic, CPU timer, DMA control, etc are on separate microprocessor chips.

The most important differences between mainframe and micro user's expectations of compatibility are in the realm of I/O device independence. Mainframe product lines are designed for long-term growth paths with portable peripherals and upward-compatible software; hence, the logical I/O subsystems are designed to function in an upward-compatible manner with several different physical CPU architectures. Users are not expected to manipulate the I/O logic, and add-on vendors emulate existing peripherals more often than supplying the software drivers for new device types.

The microcomputer user, on the other hand, is presented with a system having numerous hooks for direct interaction

IBM PC Compatibility

with basic peripherals and system software. The system software as it is currently implemented is also open to some adaptation and alteration by the user. The variety of entrances/exits to/from the IBM code and the possibility of implementing changes which IBM may later implement in a different manner means that development of a functionally plug-compatible, software-compatible personal computer is difficult. The systems cannot be identical and do not have to be but they should function as if they were.

Another point with regard to mainframe versus micro user compatibility expectations is that the micro resembles a terminal on a mainframe system more than the system itself. It is likely to expand relatively slowly, and be used until it is worn out. At the present state of the market, being able to share applications between comparable micros is of more interest than a lengthy compatible upward growth path into mainframe-level power. Many feel that the really important questions of compatibility in the system begin with a software implementation that allows buying of off-the-shelf packages that require little or no alteration to run.

☐ Board Compatibility

Can I take an IBM board or a board-level option, like the color/graphics display adapter or the expansion box, and plug it into my compatible system? Conversely, are there any alterations needed on the PC-compatible system in order to attach IBM-supplied options and peripherals?

The question of whether a PC-compatible system can add board-level options and I/O slots is easy to determine, because here we are talking about an obvious physical interface that probably would not be implemented without associated logic support. Three levels of compatibility usually can be specified. In the most compatible systems, the basic system unit can accept IBM boards, including the board that allows attachment of the IBM expansion unit for adding hard disk and additional I/O slots. Eagle, Corona, Compaq, and Columbia Data systems are examples. At the second level of compatibility, the PC-compatible system does not itself accept IBM add-on boards, but does attach a special expansion unit that permits attachment of IBM boards. This method of attachment is used by the Bytec Hyperion and Seequa Chameleon systems. In both cases the 8288 bus controller chip and the 8237 DMA controller chip or their compatible counterparts would be the likely control modules. The vendor must provide a PC-compatible bus interface to allow IBM-compatible boards to be attached.

A third level of compatibility is provided by the presence of 2 standard device ports, a Centronics printer-compatible parallel interface and an RS-232C serial interface. These are characteristic of most PCs regardless of IBM compatibility.

Of course, the presence of a compatible physical interface does not mean a device can be attached and run. A compatible I/O driver for the specific device must exist in the system, or be added.

The physical attachment of I/O devices can incur unexpected obstacles. For instance, when IBM unveiled the PC/XT with 8 slots, (as opposed to the 5 in the PC 1), the slots were positioned more closely together in the XT system unit than in the PC 1. IBM boards fit; but some boards made by third-party vendors were too thick, especially if they had a piggyback option.

Users attaching terminals to RS-232C interfaces may find that a male plug is required on one system and a female on another depending on whether the terminal is classified as a DCE or DTE communicating device by the manufacturer of the system unit. This type of physical incompatibility is easily solved with special adapter cables.

☐ Keyboard Compatibility

Is the keyboard supplied with the compatible system identical to the keyboard supplied by IBM? If there are additional keys on the compatible system, how do they affect the compatibility of programs or data that have been generated with the use of these keys?

The consideration of keyboard compatibility between the PC and its PC-compatible lookalikes has two aspects, the man-machine interface and the system compatibility interface.

The 83-key, IBM PC keyboard deviates from the familiar selectric keyboard in a number of respects that many users claim is annoying. Most irritating is that the left shift lock is not in its traditional place next to "2," but is one key over, with a backslash key included. Some users don't like the function keys placed in a pad to the left; they would prefer them to be placed in a row across the top of the keyboard. See Figure 1.

The key issue of what a single user prefers, however, is complicated further when multiple systems are used in a business setting, and people find themselves using different micros at different times. If each one has a different keyboard layout, the user's conditioned responses are snarled; error rates and irritation rise while throughput goes down.

Hence, vendors aiming at PC compatibility usually supply a keyboard identical to IBM's in spite of known problems, like that of the placement of the left shift key. If alternative layouts are supplied they are usually moves toward the standard Selectric layout. Some vendors supply more than one alternative.

This brings up the question of compatibility. To what extent are alternative layouts likely to be incompatible?

The answer is that a change in position is unlikely to cause problems; the addition of new functions may or may not. The reason for the equivocation is that IBM's method of implementing the keyboard-to-CPU communications is very amenable to software alterations. No individual key produces a single code which is translated directly to an ASCII equivalent; instead 2 codes are produced, one as the key is struck and one as it is released. The codes are collected in a 20-character buffer which is controlled by a separate microprocessor chip, the 8048, as well as ROM BIOS. BIOS processes those single keystrokes or keystroke combinations which generate legal PC codes, i.e., a 256 extended ASCII character set, plus 40 function key codes. The 8048 monitors and reports keyboard activity, gener-

IBM PC Compatibility

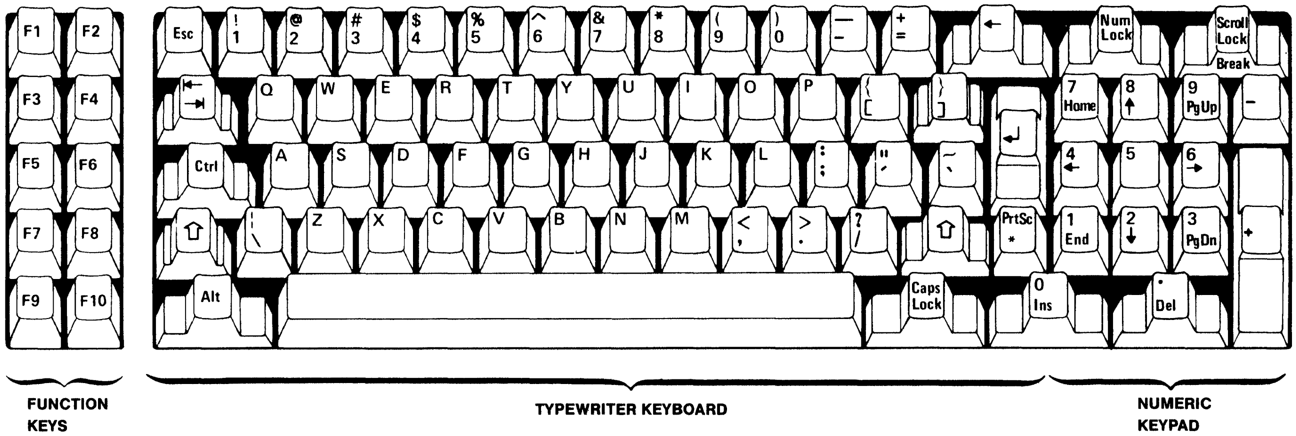


Figure 1 • IBM PC keyboard (reprinted form IBM BASIC User's Manual, pp. 2-8).

ates a BIOS interrupt for every key action, notifies the user if the buffer is full and the system isn't responding to interrupts, performs diagnostics and so on.

This indirect path between the keyboard signals and the generation of characters means that the keyboard character set can be easily redefined; changed key positions can be coded by means of software; and new function codes entered.

The problem with this flexibility is that if a software vendor makes use of it to assign unique functions to unused code combinations, the software can become incompatible if IBM assigns a standard function. The bright side is that PC-compatible vendors are not likely to have great difficulties in designing compatible keyboards or implementing a fix for a bug.

☐ Display/Graphics Compatibility

Can I take IBM programs that use the PC all points addressable (APA) graphics features and run them on my compatible system? Conversely, can I run graphics programs supplied with my compatible system (if any), and run them on the PC?

The IBM PC color/graphics board provides for all-points-addressable (APA) graphics on a color monitor. The control logic assumes a display with 640x200 picture elements (pixels).

The graphics capabilities provided by this framework are felt by many vendors to be less than the best possible with the current state-of-the-art. Consequently, there are many vendors that preserve IBM compatibility in most respects, but choose to implement a system with better but incompatible graphics resolution. Adding more pixels per square inch creates a sharper character image and plot-line, but it makes problems for software that addresses 640x200 pixels.

This type of incompatibility is common among IBM competitors that otherwise call themselves PC compatible; they provide (incompatible) enhancements. Some manufacturers have found solutions to these compatibility problems that make the best of both worlds. The Corona PC, for example, has 1280x325 pixels but addresses 640x325; each x-axis instruction parameter relates to 2 pixels, which is easily adapted to IBM compatibility. That still leaves compatibility problems on the y-axis. The logic provided by Corona manipulates all 325 pixels independently; i.e., it is incompatible.

However, the system has been designed so that a user who wants to use a program that utilizes PC-compatible graphics can add the IBM graphics board to the system; the display is then manipulated as if there were only 640x200 pixels.

☐ Disk Media & Data Compatibility

Can I take a diskette recorded by the compatible system and read it into an IBM PC? If I read it into the system, will the running program be able to manipulate the data without change or conversion first? Conversely, can I take a diskette recorded by the IBM PC and read it into the compatible system? If I succeed in reading the data, can the running program use it directly, without additions, deletions, or alterations?

Diskettes (or floppy disks or flexible disks) are the primary storage medium for business microcomputers. They have been the preferred medium because they combine the random access capabilities of the fixed disks and disk packs familiar to mini and mainframe users, together with the portability associated with reels of tape. Although cassettes and program cartridges are cheaper and have been used on small micros, they have not taken hold the way diskettes have. IBM offered cassette storage connection capabilities on the original PC, but never supplied a drive; the ability to connect a cassette drive is not offered on the PC/XT at all.

Even on systems with larger fixed hard disks, the diskette drive is used to load programs, store data offline, and back up the fixed disk. A streaming tape drive might be more logical for back-up but a tape drive is usually more expensive and not as multipurpose as the diskette drive—diskettes are more mailable and can be accessed ran-

IBM PC Compatibility

domly. Data on the diskette drive, moreover, is usually recorded in the same format or a subset of the format used on the hard disk; each of them is preformatted and is usually accessed through the same basic logic and tables that control the fixed disk. So, when a user is inquiring about disk compatibility, the user refers to both disk and diskette media, and tends to focus on the removable, exchangeable diskette.

In order for diskettes from two different systems to be fully compatible, they must be compatible in four ways. First, the physical diskette media have to be the same size. Second, the data records have to be recorded in the same machine code (ASCII, EBCDIC) and use the same basic file and record organization with the same headers and pointers. Third, the way the data track is formatted on the diskette has to fit the operational characteristics of the drives on both of the systems. Fourth, the access logic in the system and in the running program must match the organization of the directories on the diskette, or the system can hang up.

Physical Compatibility

IBM PCs store data on 5.25-inch floppy disk media; but many other systems use higher-capacity 8-inch, or more portable 3.5-inch media. The first level of diskette compatibility is easy to determine, i.e., whether both systems can read diskettes of the same size.

Code & Record Compatibility

Since the PC-DOS operating system is a variation on MicroSoft's MS-DOS, both systems use the same ASCII code set and organize records in a similar manner which is reflected in the diskette records. Headers, trailers, record delimiters, and so on, are common to both systems, and the same basic instruction types are used to manipulate files.

Recording Format

The PC-DOS and MS-DOS operating systems have comparable features for user manipulation of files, tracks, and sectors. A notable difference is that IBM's PC stores elements of the Basic I/O System, including most of the disk handling logic, in a proprietary ROM. This ROM's functions must be duplicated, preserving the same entry points, but without duplicating the IBM code exactly. This is tricky, but not impossible.

Because the read/write operations for the diskette drives are designed to be firmware and software controlled, and are accessible to the user, it is possible to manipulate the recording format and density to some extent without sacrificing compatibility. IBM has done so itself. Under DOS 1.1, diskettes are formatted at 8 sectors per track, 40 tracks per side whereas under DOS 2.0 and 2.1, there are 9 sectors per track. In both cases, 128K, 256K, 512K, or 1024K bytes can be set as the sector size; the end result is that diskettes recorded under DOS 2.1 or 2.0 can have a capacity of 360K bytes, while DOS 1.1 diskettes have a maximum of only 320K bytes. DOS 2.0 and 2.1 are designed to automatically recognize how the diskette has been formatted, and to read the diskette accordingly.

The 80186-based Pronto 16 MS-DOS system attaches diskette drives designed to record 5.25-inch diskettes at twice the number of tracks per inch as the IBM PCs. Instead of the 48-track-per-inch PC spacing (with 40 tracks for recording data), the Pronto 16 is designed to record at a 96-track-per-inch spacing (with 80 tracks for recording data). The controller identifies the 96-track-per-inch recording as a double-density mode, and can also record in a single-density, 48-track-per-inch mode. By skipping the odd-numbered tracks, the controller can read or write in PC-compatible format; the manufacturer can provide twice the capacity without sacrificing IBM media compatibility. The Pronto 16 user can read IBM diskettes without difficulty "all other things being equal", and can write equally compatible diskettes by using single-density mode. However, if the double-density mode is used, that diskette cannot be read into an IBM system; it will have to be copied onto 2 single-density diskettes.

PC-DOS 2.0/2.1 & Its Directories

Directories for access of the contents of each diskette and disk reside on that diskette or disk. The logic for control of the directories is included in generic MS-DOS-BIOS, but in PC-DOS, most of it resides in IBM's proprietary BIOS ROM. DOS 1.1, equivalent to MS-DOS 1.25, is oriented toward systems with diskette storage but no hard disk. Diskette files are accessed through a single directory; an adequate access structure for that medium. PC-DOS 2.0 and 2.1, as well as MS-DOS 2.0 and 2.05, on the other hand, supply a combined partitioning and a hierarchical tree structure for disk access. The partitioning allows a disk to be divided among multiple operating systems. The tree structure provides for multiple sub-directories, and can define a variety of access paths through the directories. This routing mechanism allows a sharing of common code and data (with some limitations), and a flexible upper limit to the virtual partition size created by the path. Unfortunately software designed for PC-DOS 1.1, which includes most of the third-party PC software currently being sold, will not recognize pointers that lead to sub-directories. The program often can run on the hard disk, but must use the single-level type of directory, which is usually unwieldy for a storage medium the size of the fixed disk. Furthermore, if files needed by the program are defined as being accessed only by a sub-directory rather than the current directory where the program resides, the program will hang up since it cannot access by means of the sub-directory.

PC-DOS 2.0 and 2.1 can run on diskette-based systems using the same ROM chip originally delivered for the PC 1. However, if an expansion unit with hard disk is added to the system, one of the elements of the installation package is a new ROM chip with the fixed disk driver on it. This chip will undoubtedly present compatibility problems as yet unknown to us, since XT-compatible systems are just emerging.

Rules of Thumb

The changes in file access methods between the operating systems versions complicates the compatibility picture for diskette media. Previously, a user would have a fair degree of assurance a compatible diskette could be recognized. When the basic system ran MS-DOS on an 8088, 8086, 80188, or 80186 CPU, and recorded on 5.25-inch disks

IBM PC Compatibility

with a 320K-byte capacity, the fair assumption could be made that if the manufacturer claimed compatibility, the disk/data would be compatible with a PC-DOS system. This would be particulary true if the non-IBM vendor claimed to be able to read PC-DOS into the system and run under it.

Now however, the question arises, which IBM system running under which version of PC-DOS? The need to specify which version of programs and systems is compatible will become more acute as software vendors begin to issue programs that can utilize IBM DOS 2.1 features.

The addition of an expansion box with a fixed disk to a PC-DOS 2.0 or 2.1 system adds another wrinkle to the compatibility picture because IBM supplies a new ROM chip that includes the fixed disk drivers. This means that PC-compatible vendors cannot develop new DOS 2.0 or 2.1 equivalent that support fixed disks without studying the new ROM, even though diskette-based DOS 2.0/2.1 can run the original PCs. It further means that a user could buy the expansion box/fixed disk and plug it into a PC-compatible system, but it would not work unless it was running under the IBM-supplied chip. Third-party vendors supply fixed disks with a software driver that run in conjunction with the original PC ROM chip rather than the new one.

■ MAKING PRELIMINARY ASSESSMENTS OF COMPATIBILITY FEATURES

The potential user of a microcomputer is faced with a marketplace which is too young to have developed comprehensive clear-cut standards or even many traditions that serve as substitutes for formal standards. In some cases, de facto standards have approached the equivalent of a formal standard due to widespread implementation. Most systems have a (standard) RS-232C interface and a Centronics-compatible parallel printer port, for instance. On the other hand CP/M, MS-DOS and UNIX generic operating systems may be highly portable, but their implementations are hardly standardized. Following are descriptions of typical enhancements over PC specifications that vendors might be expected to implement, and types of PC programs that are most likely to present compatibility problems on a PC-compatible system.

□ Typical Enhancements & Their Effect on Compatibility

Compatible vendors at any level of the market must offer the user price advantages and equivalent or better reliability versus the IBM PC if they are to stay in business. The most successful compatible vendors usually offer more prompt delivery and more responsive service than the target vendor.

All PC-compatible vendors are also aware that they need to do more than this in order to provide a significant identity to their product, and offer features to the user that are additional attractions. Some of these features do not affect compatibility; others might. The following discussion of typical PC-compatible enhancements is focussed on their impact on compatibility.

• **Portability or Other Physical Packaging Alterations.** A number of PC-compatible vendors offer portable systems complete with display, diskette drive(s), keyboard and system unit, but not usually including a printer. This feature need not affect compatibility at all, nor does any other alteration to the physical housing

• **Improved Ergonometric Display Design.** The background, color and manipulative characteristics of the display (tilt and swivel), additional nonglare features and physical size of the screen do not affect compatibility.

• **Increased Resolution of Display.** As explained earlier, the display resolution can be improved by increasing the number of picture elements (pixels) per square inch. This increases the distinctness of alphabetic and numeric character shapes. However, doing so can and usually does affect compatibility. If the change involves a fixed formula that automatically changes IBM's 640x200-point addressable graphics matrix (into a 1280x400 or 1280x600 matrix, for example) it can be manipulated. This is explained at greater length earlier, and is an important point for evaluating compatibility claims.

• **Increasing CPU Clock Speed.** This practice is fairly common; it only creates problems for time-dependent programs.

• **Increased Memory Capacity.** Increasing the memory in the basic system does not change the compatibility of a vendors product, but increasing memory beyond IBM's limit might do so. The PC address structure allows addressing up to 1M bytes, but some of those addresses are used for system functions that are not supposed to be accessed by the user directly. The 640K-byte PC limit for user memory is large enough for a single user-single tasking system that most alternative vendors do not exceed it anyway.

• **Increasing Diskette Capacity.** The PC line handles 160K, 320K, or 360K bytes of diskette capacity depending on the drive model and operating system release. In almost all cases any change in diskette size, capacity, file handling, formatting, or directories does affect media compatibility. The exception is when a drive and its supporting software are engineered to read and write PC densities as well as the larger density.

• **Changes in Keyboard Design.** Changes in a keyboard's physical characteristics, such as touch sensitivity, shape of keyboard top, color coding of key caps, labelling of key caps, positioning of keys, movability and tiltability of the keyboard unit, do not affect compatibility. Adding function keys may or may not, and probably won't because they are entered into the system by a two-level process that allows software manipulation and translation of keyboard input into standard system instructions. See the previous discussion of the keyboard, as well as the discussion on typical software problems.

• **Adding to the Number of Slots, RS-232C, or Parallel Printer Ports.** In and of themselves, these changes do not create problems, provided the devices can be addressed. There may be problems with software that directly addresses machine elements.

• **Combining Logic Cards.** Many PC-compatible vendors provide a display control card which combines the func-

IBM PC Compatibility

tions of the monochrome display control card and the graphics display control card, which are separate for all PC systems except the 3270 PC. This combination does not affect compatibility, and saves I/O slot positions; in fact this "compaction" is a favorite device of vendors competing with IBM.

• **Implementing a RAM Disk.** A RAM disk is a portion of memory set aside to be treated as if it were a diskette in a disk drive, hence speeding up all operations that involve diskette access. PC-DOS 2.0 and 2.1 as well as MS-DOS 2.0 implement a RAM disk but PC-DOS 1.1 and MS-DOS 1.25 do not. Certain PC-compatible vendors include RAM disk capability in their version of MS-DOS 1.25. The addition of this capability may or may not affect compatibility with programs designed to run under PC-DOS 1.1; in the case of Columbia Data, and Corona systems, for instance, it does not.

□ Types of Programs Likely to Present Compatibility Problems

The purchase of programs is an ongoing process. When a user owns a system that is PC compatible, it is still necessary to test software designed to be PC compatible. The following presents some items to look for in a program to try to determine in advance whether or not it is going to be incompatible.

• **Programs Using the APA Graphics Facility.** IBM's graphics capabilities leave something to be desired, and hence are a favorite point of departure for vendors who are partially compatible with IBM. Spreadsheets, games, and other graphics-related programs need to be investigated carefully if the user is considering running them on a machine that is sold as IBM PC compatible. If the PC-compatible machine uses an IBM graphics board, however, programs will probably run. See the previous discussion of graphics.

• **Programs Which Directly Address ROM.** Although both IBM PC-DOS and Microsoft MS-DOS programming guidelines disallow direct ROM calls, specifications on the interrupt system and assembly language implementation of the PC provide plenty of information on how to do it. As a result, programs written with direct ROM calls imbedded in them are likely to hang up even on the most compatible of systems, since IBM's ROM is proprietary. Fortunately, most large business software vendors are interested in making their programs as portable as possible, and consequently tend to avoid ROM calls. They are more likely to occur in programs which enhance or supplement a particular PC function, or in games programs.

• **Programs Which Directly Address PC Machine Components.** As stated earlier, IBM provides elaborate details on the interrupt system; this means that programmers can directly address machine elements in their programs. This is not a problem if the PC-compatible vendor uses exactly the same chips to implement the DMA controller, the timer, the keyboard controller, the serial I/O controller, etc. Each of these controllers is actually a special-purpose microprocessor with its own instruction set. However, if the manufacturer chooses another chip to implement more capabilities, these types of programs may not run. For example, the Bytec Hyperion uses a Zilog serial I/O chip that uses different commands than IBM's asynchronous control card. This means that PC programs like Hostcomm and PC-Talk will not run on the Hyperion, because they are designed to function with the IBM card.

■ CURRENT COMPATIBILITY LEVELS

Because the IBM PC-compatible marketplace is so young, an accepted set of distinctions for levels of compatibility have not yet come into widespread use. Users and industry analysts tend to divide the market up into three broad divisions. It is assumed in all three divisions that the systems run some version of MS-DOS and are based on an 8088, 8086, 80186, or other compatible member of the Intel 8088-related family of microprocessors. These divisions are as follows: MS-DOS compatible only, disk/data compatible, and PC compatible.

□ MS-DOS Compatible Only

This broad group of systems runs MS-DOS but makes no particular effort to be IBM PC-compatible. Certain programs which run on the PC can run on these systems because they use only the code common to MS-DOS and PC-DOS; in other words they are highly device independent. They will not directly address ROM and usually do not use graphics, for example. Because there are PC programs which do run on the system, the vendor may claim "some" IBM compatibility. Usually, however, the specifications for these systems show differences in diskette media size or capacity (not 5.25 inches and 160K/320K/360K bytes), differences in graphics pixels (not 640x200 for monochrome and 320x200 for color), a different keyboard, different character capacity and format on displays (not 24 lines of 80 characters each). These systems do not attach IBM peripheral cards because if these vendors were interested in that degree of IBM PC compatibility they would have included disk/data compatibility—which is easier to implement.

This level of compatibility has been labelled as "MS-DOS compatible," or "incompatible," by industry observers. Only a stray manufacturer or two calls the system "PC compatible" if it exhibits this level of compatibility, because most manufacturers of this type of system see a better marketing advantage in having a distinct identity. These vendors will usually say that they are not making a "me-too" product.

□ Disk/Data Compatible

The second large group of systems can read and write disks in the same format used by the PC, and use the same physical size diskette. They will produce 5.25-inch diskettes with a capacity of 160K, 320K, or 360K bytes depending on the MS-DOS version implemented and whether or not the diskette is single sided or double sided. Sectoring will be the same, i.e., 8 or 9 sectors per track and 128K, 256K, 512K, or 1024 bytes per sector. Headers, trailers and directories will be the same, with the qualification that certain compatibility problems will arise between diskettes with multiple sub-directories recorded under MS-

IBM PC Compatibility

DOS 2.0 or PC-DOS 2.0/2.1 and single directories recorded under MS-DOS 1.25 or PC-DOS 1.1. See the previous discussion on disks for more details on this problem, which affects PC systems as well.

This level of compatibility has been labelled as "disk compatible," "data compatible," or " media compatible," and of course, "PC compatible." It is a level of compatibility which is fairly easy to implement because the basic software structures are already in place in MS-DOS. The manufacturer just needs to add a 5.25-inch diskette drive and some coding to implement this type of compatibility in addition to the features which are already on the system. Furthermore, this level of compatibility is easy to test for, except for the impediments posed by the directory problem—which is characteristic of IBM PC users as well as PC-compatible systems users.

☐ PC-Compatible Systems

Systems which are fully or almost fully PC compatible with existing hardware and software include the capability of attaching PC expansion boards and running PC software. The keyboard may be identical as well as compatible; the disk is compatible, the system runs most non-graphics and some graphics software and usually can run interpretive BASIC with graphics as well as compiled BASIC. The system can run PC-DOS 1.1, as well as PC-DOS 2.0 and 2.1, and CP/M-86. If the system does not quite meet these standards, the point of variance usually is with the graphics capabilities and the ability to run interpretive BASIC with graphics.

By now the reader knows that since PC-DOS has a lot of capability for supporting machine-dependent features, and even the most compatible system could hang up on a program that directly calls ROM, perhaps even if the compatible system has all ROM entry points at the same locations as IBM's. It is also possible for a PC-compatible system vendor to have designed the ROM code without violating IBM copyrights and yet function in the same manner. Only testing will tell.

Six systems have been on the market long enough to gain a reputation as to compatibility levels. The Compaq system, the first to achieve a sizeable installed base, is considered to be the compatibility standard. We have heard of no reports of a PC-DOS 1.1 program that did not also run on the Compaq system. Columbia Data and Corona systems have also achieved excellent reports on compatibility level. The Corona system has incompatible graphics features, but can operate in a fully compatible graphics mode if the IBM graphics board is installed.

The Bytec Hyperion, Eagle PC, and Seequa Chameleon are also largely IBM compatible. The Bytec and Seequa systems both require a vendor-supplied expansion box to attach IBM boards. The Bytec and Eagle systems also have different keyboard layouts and additonal function keys, but as explained earlier this does not affect software compatibility. All three have some software incompatibility; the Bytec Hyperion does not support IBM graphic software, the Seequa Chameleon does not support Interpretive BASIC with graphics; and the Eagle appears to have problems with Interpretive BASIC, a fair amount of graphics software and some non-graphics software.

This level of compatibility has been labelled "system compatibility", "operational compatibility," "truly compatible," and of course we have again, "PC-compatible."

Depending on how compatibility levels are defined, there are a number of systems that fall into grey areas. For instance some industry periodicals refer to a category called Functionally Compatible. The Functionally Compatible category (which by the beginning of 1984 included only one member, the TI Professional), includes Data Compatible (i.e. disk/data compatible) systems which run their own versions of a large majority of top IBM programs, and thus achieve functionality comparable to the IBM PC. The TI system, however, does not attach IBM peripheral cards, or make claims of overall IBM compatibility. Obviously, this category is a little soft, because it is open to interpretation as to when a disk-/data-compatible system running under MS-DOS has adapted enough of the major software to fit the "Functionally Compatible" label.

The "PC Compatible" label as we have described it does have a handy dividing line for separating the sheep from the goats: the ability to attach IBM boards either directly in the system unit or by means of an expansion box. If a vendor has gone to the lengths required to implement this feature, there are usually a number of other features on the system that broaden the available compatible software base as well.

■ CONCLUSION

The definitions of PC compatibility include MS-DOS systems which are disk/data compatible and those which can attach PC boards; both these levels of compatibility can run more PC software than systems supporting MS-DOS compatibility. The system commonly accepted as being the most compatible system, Compaq, does not claim to be—but appears to be—fully compatible except for BASICA, because users report they can run PC-DOS itself and IBM-supplied system software.

However, it would be wise to test every program the user wants, even on a Compaq system. The two favorite testing vehicles are MicroSoft's Flight simulator, and Lotus 1-2-3.

• END

Microprocessor Chip Technology

Understanding the Significance of the Chip Used as the Heart of the Microcomputer

■ INTRODUCTION

Every time a new microprocessor chip is announced, the trade press is full of promises that the chip will revolutionize the industry or provide stiff competition for existing favorites. Yet many products universally hailed as technical milestones seem never to catch on, while others greeted by "HO, HUM" become new standards. Even the basic question of "is 32 bits better than 16, and 16 better than 8" seems to have no clear answer. This report examines the microprocessor, the products available, and the importance of microprocessor features in the selection of a microcomputer.

WHY TALK ABOUT COMPUTERS AS IF THEY WERE PEOPLE?

People who do not work with computers daily are sometimes amused by the fact that those who do almost unconciously talk about computers as though they actually had intelligence, will, mercy, and spite. The more closely one works with computers, the more pronounced this trend is.

Computers do not have and perhaps never will have actual intelligence; "HAL" in the movie "2001" is probably not due to happen until at least 2001. Saying "the microprocessor knows that the memory..." does not indicate that the speaker really believes that the microprocessor has intelligence and awareness in a human sense (at least, not usually). Computer specialists and users often adopt that anthropomorphic form to convey a more complex and more wordy thought. Thus, we might be technically correct in saying that "the microprocessor has an associated control line which is used by the memory system to indicate that the address requested by the microprocessor is not part of the available memory space." Or, we can say "the microprocessor knows that part of memory isn't there."

With no apologies to those who think that attributing human traits to a computer dehumanizes the rest of us, and in the interest of clarity of thought, we will use the latter form throughout this report.

MICROPROCESSORS VERSUS MICROCOMPUTERS

One of the very first problems which personal computer buyers may face is one of terminology. What is a microcomputer? The same thing as a microprocessor? If not, what exactly is the difference?

Microprocessors are electronic components; not computers. Often called "microprocessor chips" to make their status more obvious, the average product is about two inches long, less than an inch wide, and may cost less than 10 dollars. Inside the chip are microelectronic circuits which permit the microprocessor to recognize instructions and move information.

Most microprocessor chips are totally useless alone. They require an external source of memory in which to store instructions (collections of which are called "programs") and the information on which they are working. Additional devices provide the ability to read from and write to disk drives for long-term data storage, and to read operator requests from a keyboard and display instructions on a CRT or video monitor. The collection of these parts is called a microcomputer. Figure 1 shows a block diagram of a microcomputer, including the microprocessor, memory, and device controllers.

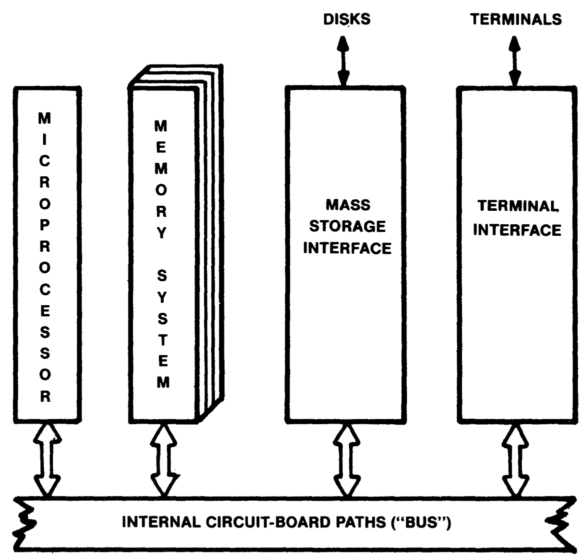


Figure 1 • block diagram of a microprocessor.

Understanding the workings of a microprocessor requires a prior understanding of how the computer which contains it must function. This is something which can be best acquired by starting with the memory of the system. Figure 2 shows computer memory as a series of "bins," each of which can be identified by a sequentially assigned number. Any bin can be used by the system as long as its number, or "address," is known. The capacity of a "bin" is usually a single data character, known in computer jargon as a "byte." A byte is a binary number which can represent a single letter, number, punctuation symbol, etc. Most microcomputers use 8 binary digits (bits) to represent a byte, so a little binary math shows that there are 256 possible values for any single "bin" or memory position. Figure 3 shows the relationship between "bits," "bytes," and normal letters, numbers and punctuation symbols.

Microprocessor Chip Technology

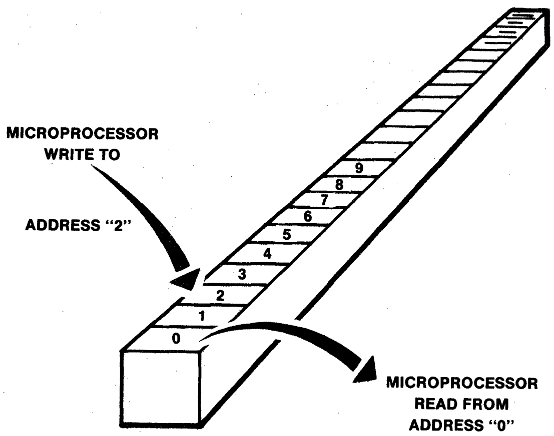


Figure 2 • memory and addressing.

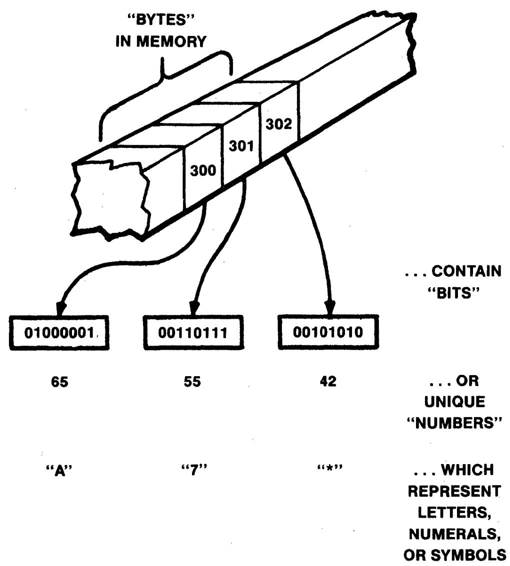


Figure 3 • bits, bytes, and data.

Memory can be used to store data which is being used at the time, but it is also used to store instructions for the computer to execute; the program. Anyone who has used a complex hand-held calculator has had some experience with programming. For example, a problem in a typical business might be to take the wholesale cost for an item, add a fixed handling cost, a markup expressed as a percentage, and thus calculate the retail price. On the calculator, this is a quick: COST + FIXED HANDLING X MARK UP = PRICE.

If you have to do a couple hundred this way, however, it isn't so quick. A computer is a calculator that remembers instructions. A program, developed by a specialist called a "programmer" provides the computer a definition of the task, and generally a place where the input information is to be found and the results are to be written. In a popular commercial programming language called COBOL, our calculator cost/price problem would look like this: COMPUTE PRICE = (COST + FIXED-HANDLING) • MARK-UP.

The above operation is complex, and would actually consist of several steps, as it did in the calculator example. Furthermore, computers do not directly use languages such as COBOL, which are designed for easy human use. Rather, such languages (called "high-level" languages because each of their single steps may translate to many computer steps) are converted to the computer's own language by another program called a "compiler." The compiler reads the programmer definition of an instruction, such as the COMPUTE in the example above, and converts it to the machine instructions needed to perform the operation. Each step would require one or more "machine instructions" to define it, and each instruction, depending on its complexity, would require one or more memory positions to hold it. Programs are stored in the computer's memory, along with any data the program might need. What actually performs the computations, or "runs the programs," is the microprocessor. Microprocessors have a specific number of "instructions" which each will recognize and perform, and the collection of such instructions is called the "instruction set" of the computer.

MICROPROCESSOR OPERATION & STRUCTURE

A microprocessor operation, the execution of an instruction, can now be defined, using our memory concept in Figure 1, as follows:

1. The microprocessor asks memory to deliver an instruction for execution. This requires telling memory where that instruction is located, one position at a time, and reading each position of the instruction until the entire instruction has been acquired.

2. The microcomputer evaluates the instruction and determines what must be done. The instruction includes the location of the data to be used, and the place where the results are to be stored.

3. The source data is read from memory, in the same way the instruction was read.

4. The requested operation is performed.

5. The results of the operation are written to memory. Again, this involves telling memory where the data is to be placed, then writing the data to that location.

The example of microcomputer instruction execution given above is, in some ways, simplistic. It illustrates the way in which memory and the microprocessor interact, and on analysis will demonstrate the main elements of a microprocessor.

The requirement that the computer identify a memory location means that the microprocessor chip must provide a memory address for any read or write operation to memory. This is most often accomplished by providing the microprocessor with a series of address lines, electronic connections which the microprocessor can send signals on to identify memory locations. Computers operate in binary

Microprocessor Chip Technology

(on/off) arithmetic, so the chip will need quite a few address lines to address a large amount of memory. Figure 4 shows the connection diagram for the popular Zilog Z80 microprocessor. The connections marked A0 through A15 are the 16 address lines, and the fact that there are 16 lines and thus 16 binary digits in the address means that the chip can address 2 to the 16th power or 65,536 positions of memory.

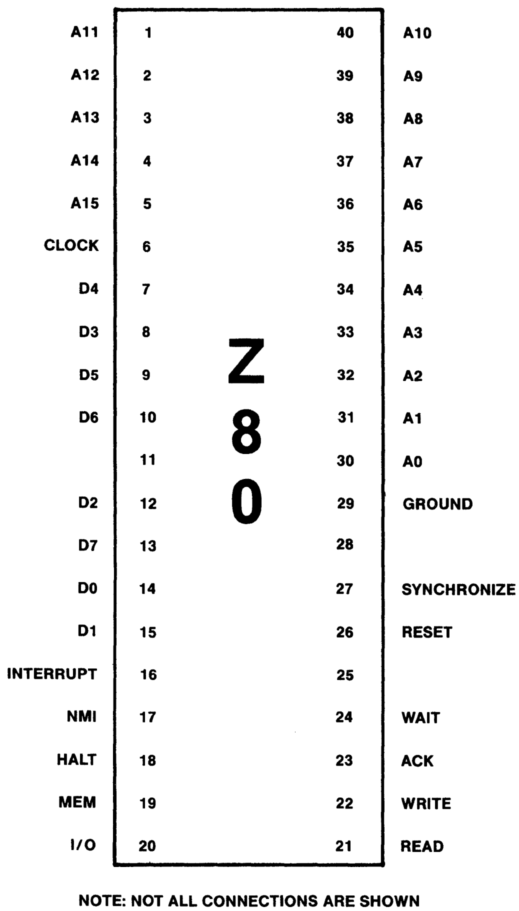
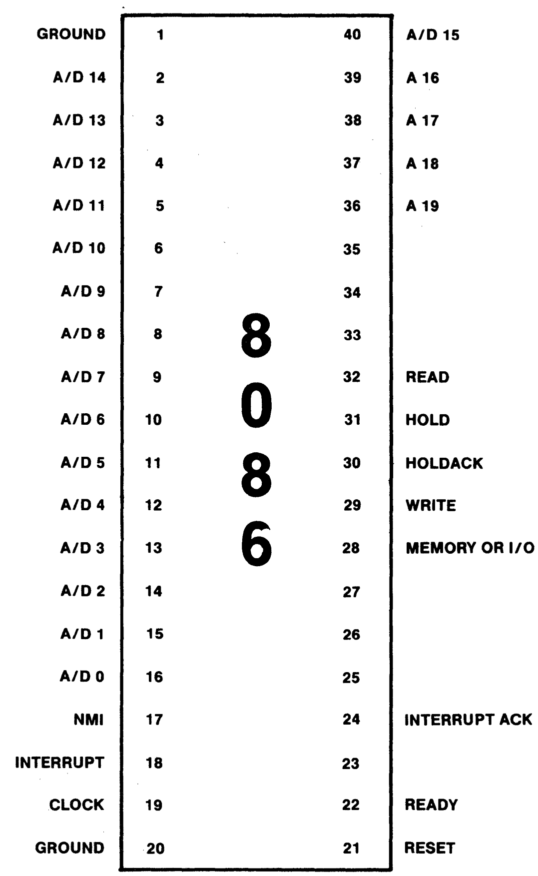


Figure 4 • Z80 connections.

Once we can address memory, we must also tell memory whether the use we intend is read or write. This can be done with a single connection, since read and write are a binary choice (2 alternatives). This "R/W" connection is also shown on Figure 4.

The data to be read or written is another issue. We have 8 binary digits in a single location, so we will need at least 8 connections, or data lines, to read or write them. As Figure 4 shows, these are labelled D0 through D7.

With the labeling of the data lines, we reach a point where we can address the meaning of "16-bit" or "8-bit" computers. Figure 5 shows the connection diagram of an Intel 8086 microprocessor. Here we can identify two new concepts. First, to save on space the chip uses the same lines for address and data (after all, we can tell memory the location and write or read the data as two steps). This is called multiplexing, and is common with the newer microprocessor chips. The other concept is that there are more data lines—lines D0 through D15. This makes the 8086 a 16-bit microprocessor—it has 16 data lines and can thus handle 16 bits of data, or two characters, each time it reads or writes memory.



NOTE: NOT ALL CONNECTIONS ARE SHOWN

Figure 5 • connections for an Intel 8086.

Having looked at the connections outside the microprocessor, let's go inside for a moment. Figure 6 shows the internal structure of a typical microprocessor chip. An Execution Unit performs the actual instructions, and contains the actual Arithmetic and Logic Unit (ALU) and a set of "scratch pad" areas where the ALU stores data while it is working with it. These are called "registers." The EU connects to the outside world through an Interface Unit (IU), which handles the data and address lines to memory, etc.

The example of an instruction execution given earlier can now be restated in terms of the structure of the microprocessor chip and its key external "lines"

Microprocessor Chip Technology

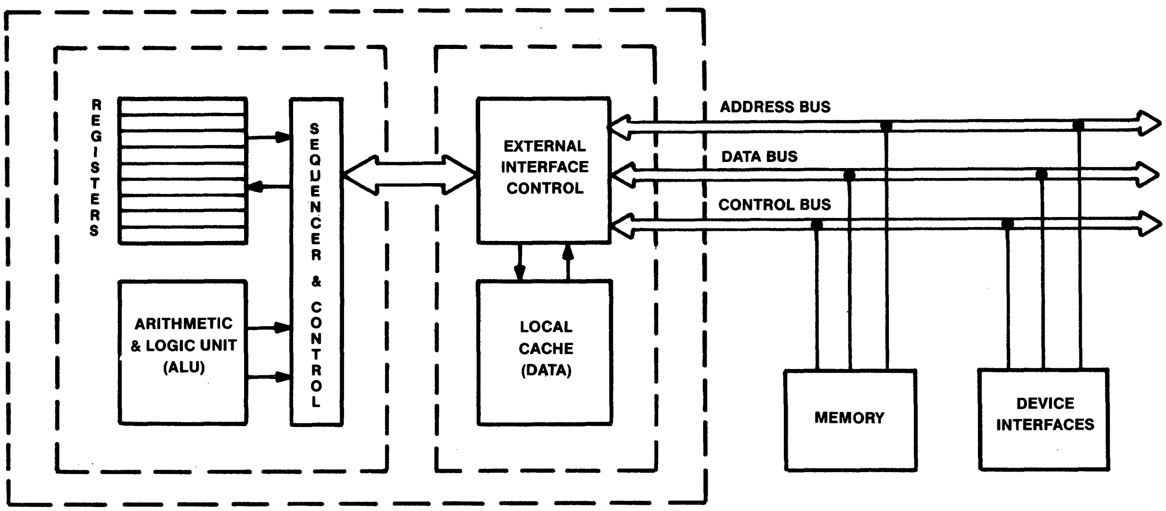


Figure 6 • block diagram of a microprocessor.

1. The ALU recognizes that it has no instruction active, and prepares a request for one. It takes from one of its registers the location of the next instruction and passes that location to the IU with a request to read the data.

2. The IU applies the binary address given by the ALU to the address lines of the microprocessor chip. These lines are also connected to memory as shown in Figure 6. Because this is a read operation, the IU sets the R/W line to its R value.

3. Memory selects the location addressed by the settings of the address lines.

4. The IU ceases signalling on the address lines, since memory now has the location and since the data will share the same connections.

5. Memory, given a read signal on the R/W line, applies the full 16 bits of two memory locations beginning with the one addressed to the data lines.

6. The IU reads the data from the lines into an internal register. If the ALU can use all 16 bits (two bytes) of data, it will deliver it to the ALU. If not, the second 8 bits, representing the contents of the higher and unrequested location, will be saved to see if the ALU requests it next. If it does, the second 8 bits can be delivered without a memory access.

7. Memory ceases to signal the data values on the data lines, leaving the lines unused.

In this more complex picture, the relationship between the microprocessor elements and the operation of memory can be seen more clearly. The example also demonstrates a fact about the 16-bit (and 32-bit) processors. Not all computer operations can use more than one character of data at a time. The various types of microprocessor chips handle the procedure differently, but certain types of programs have been shown to take only small advantage of the 16-bit data paths in processors.

CONTROLLING THE MICROPROCESSOR CHIP & ITS SURROUNDINGS

In the example shown above, there are several points where one element of the computer system places a signal on the address or data lines and expects the other element to be waiting for it. Likewise, the signals are dropped at other points, assuming that they ar received and understood. How can we be sure?

Most microprocessor chips, and the other computer elements such as memory which interface with them, rely on timing to synchronize their accesses to shared data, address, and control lines. The basic timing is supplied by a clock signal, which is another control line on the microprocessor chip. The microprocessor may generate the clock itself, or may require that it be supplied by a simple circuit elsewhere. A clock signal can be viewed as a coxswain on a rowing crew shouting "stroke . . . stroke . . . stroke . . ." Each clock "tick" can be used as a signal to do something. The IU may thus signal address information beginning at a tick and ending at the end of the tick, and if memory is designed to read the address in this same space, the information will pass correctly.

The obvious problem with this analogy is that the IU may faithfully begin its action at the start of a tick, but there are a lot of ticks. If memory thinks something else is supposed to be happening on this particular tick, we have again lost synchronization.

The computer designer solves this problem by having the microprocessor generate one or more other control signals. For example, the microprocessor could signal at the start of the entire instruction process. If the sequence of events in

Microprocessor Chip Technology

executing an instruction always stays pretty much the same, this technique will operate well.

The coxswain on a crew can affect the speed of the boat by changing the pace of "stroke" commands, and computer speeds can be changed by changing the speed, or frequency, of the clock. Clock rate is always given in terms of millions of "ticks" or cycles per second, a unit called a "Mega-Hertz" in the industry, and abbreviated MHz. Each microprocessor chip has a maximum clock rate which it can support, and most chips have models with various maximums. The Zilog Z80 has a 2-MHz clock limit, while the Z80A will support a 4-MHz clock and the B model a 6-MHz clock.

For a given processor, an increase in the clock rate will generally cause a corresponding increase in the performance of the processor. For tasks which do not involve the accessing of external resources such as terminals or disk files. There are some factors in the design of computers which may affect this, however. One is the ability of the memory elements of the computer to keep up with the clock speed of the microprocessor chip. Many microprocessors can operate very fast using the internal registers and so may want to run at clock speeds of much greater than 6 to 8 MHz. At these high speeds, memory to keep up may be expensive or unavailable, so the designers of the computers build in delays at the points where the microprocessor chip and the memory components must interact. These are called "wait states," and they have a significant effect on performance if they are employed. An 8086 computer system advertised as operating at "8 MHz with no wait states" has memory components fast enough to permit the use of an 8-MHz clock without introducing delays. Each wait state introduced adds one tick of the clock to the operation affected. The exact effect of wait states on performance depends on the clock speed, the number of states, and the design of the microprocessor.

Clock speed is not a good way to measure the relative performance of different microprocessor chips, and therefore not a good way to compare different microcomputers. The reason is that many chips with very simple instruction execution capabilities can do much more in a few ticks of the clock than a complex chip which provides almost minicomputer or mainframe instruction power. For example, a 6502 microprocessor chip (one of the very oldest types still used commercially) operating at 2 MHz will outperform an 8088 (the type used in the IBM PC), and operating at 6 MHz in certain types of operations.

THE ALU & INSTRUCTION SETS

The ALU portion of the microprocessor chip, as we have said, is the element of the system which executes ("follows") the instructions written by a programmer to complete a task. The instructions which a computer recognizes are collectively known as its instruction set, and the size and makeup of an instruction set will affect the way in which a computer can be used.

An instruction in a computer may take several forms, but there are some common elements, as shown in Figure 7. All instructions must have, and usually must start with, an operation code, or op-code. This defines what the computer is actually expected to do. Following the op-code are one or more addresses which locate the data needed for the operation specified. Most instructions will have a pair of addresses, one defining a source and the other a destination, or both defining elements which are to be combined. These are usually called operands. Examples of the op-code/operands structure are "move A to B" or "add A to B." There are some instructions which have only one address because the operation needs only one; "increment A (by one)."

Nearly any commercial microprocessor chip can be made to complete any computational or manipulative task which a programmer may wish to define. Complex functions on microprocessors with simple instruction sets may be handled by using a series of the simple instructions to replace one more sophisticated one. For example, a computer without a "multiply" instruction can be made to multiply using an add-and-shift technique similar to that used by human beings when they multiply multidigit numbers on paper. But the substitution of multiple basic instructions for one more complex instruction will normally result in a considerable penalty in performance. In general, the more instructions a computer has, the more efficiently it will perform the kind of complex tasks which are common in business programming applications, but some computers call an "instruction" any variation on a basic operation. For example, they may count an instruction to move a character from memory to internal register "A" as an instruction, and the same operation to internal register "B" as another.

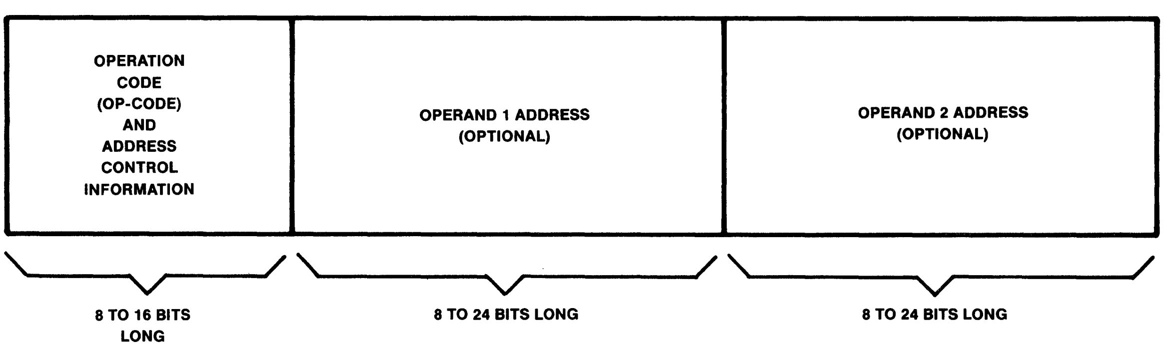


Figure 7 • the computer "instruction."

Microprocessor Chip Technology

A safer way to evaluate instruction sets is to look for specific types of instructions, ones which can make the programming of the system easier and faster and thus result in improved programs being developed for business use. The key instruction groups are:

1. Multiply and Divide instructions. The number of simple arithmetic instructions needed to simulate a single multiply or divide is considerable. Programs which are used for scientific, engineering, or statistical applications will benefit most from a built-in multiply and divide, but spreadsheet programs may do so also.

2. String instructions. Many computers operate on strings of data such as a person's name one character at a time. The ability to move data around in memory as a group of characters reduces programming effort and cuts down on the number of instructions needed to handle the kind of alphabetic information that makes up much of the business applications.

3. Register saving and restoring. This is a complex area of computer operation, but an important one. When the microprocessor is working on one task, it may have to interrupt its work to handle some special event, like the reading of a character generated by the operator pressing a key on the keyboard. The microprocessor must "remember" where it was, and that involves saving those internal registers discussed earlier. The more instructions it takes to do this, the more the computer is slowed by the interruption. In applications where data communication lines are being used, the time it takes for the microprocessor to "context switch," as it is called, may be the single most important aspect of the system.

4. Floating-point instructions. Most business calculations are performed with a form of mathematics which supplies an exact answer, to the number of decimal points required. This form of calculation is not normally supported directly by computer instructions, because there are so many variables involved (the size of the numbers, the location of the decimal, rounding, truncating, etc). Many scientific and engineering calculations need not be this precise; an answer which is accurate to four or more significant figures is sufficient. There is a special way to represent numbers such as this, called "scientific notation." Many of those who remember slide rules will remember this technique, where the number "4,078,341" would be approximated as "four point zero eight times ten to the sixth power." A similar technique in representation, called "floating point," can be used on computers. It not only allows the storage of these approximate-form numbers in less memory, it can permit much faster calculations. Scientific programs almost always use floating point, and programming "languages" designed for scientists and engineers (such as FORmula TRANslation, or FORTRAN) rely heavily on it. If a computer is to be used for this type of application, a microprocessor with floating-point instruction capability is certain to improve performance.

5. Packed decimal instructions. These are supplied with some microprocessors to support more conventional business mathematics, where a large number with three decimal places of accuracy must be manipulated. While some high-level languages such as COBOL provide high-precision decimal calculation, they do so by generating a lot of machine instructions, making the process very slow. If a microprocessor has packed decimal instructions available, it can be much faster at business arithmetic.

Some computers have instruction sets which can be extended through the use of a co-processor. The most common example of this is the Intel 8086 and its popular 8087 math co-processor. Figure 8 shows the logical relationship between a co-processor and the primary processor. The two microprocessors both have access to memory

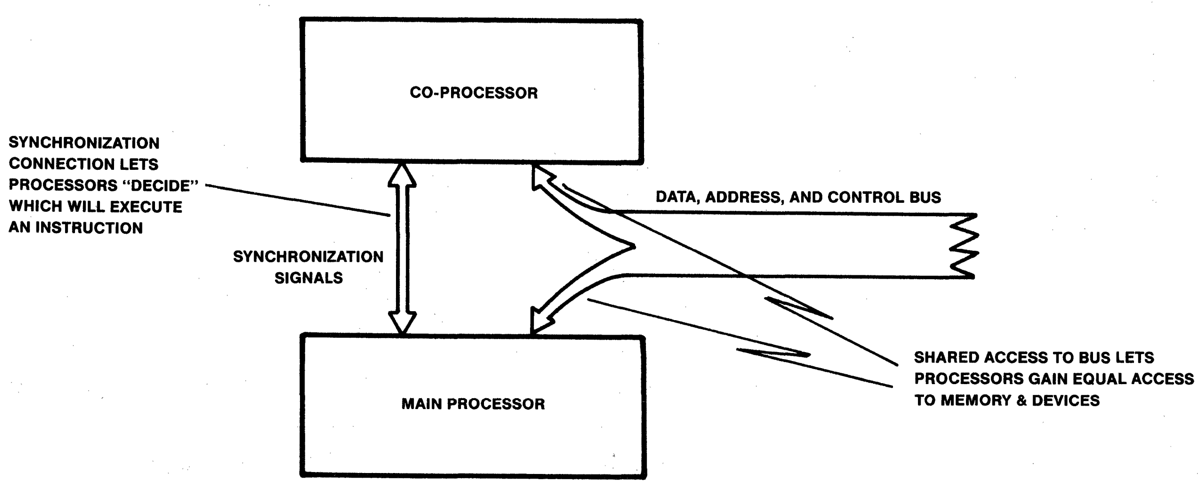


Figure 8 • use of a co-processor.

Microprocessor Chip Technology

and other elements of the system, but they cooperate in handling instructions, so that each will handle those for which it is designed. The advantage of the use of a co-processor over a single microprocessor lies in cost and complexity. Users with no need for floating-point operations may not want to pay for their inclusion in an instruction set.

ADDRESS SIZE, ADDRESS LINES, ADDRESS SPACE & COMPUTER MEMORY SIZE

The descriptions of computer operation and instruction types constantly refer to the interplay of the microprocessor and the memory within the computer. In order for the microprocessor to read or write data, it must identify the location which is to be affected. This is done by one or more operands in the instruction, and the size of the operands limits the size of the number they can represent, and therefore the size of the memory addresses. This address size is our first memory limitation. The size of the operands in an instruction limit the number of memory positions which, at any one time, can be directly referenced by the microprocessor.

We've already mentioned that the address lines are used for communicating this address to the outside world. It follows that the number of address lines available affects the number of memory locations which can be designated. A system with one address line, which can have only a zero or one signal on it since it's a binary line, can address only two positions of memory. One with two lines can handle 4, which it would know as 00, 01, 10, and 11. Three lines gives 8, and so forth. The size of the "address space" of a computer, the number of memory positions which it could use, is thus the number two raised to the power of the number of address lines. Since a Z80 computer, shown in Figure 4, has 16 address lines it has an address space of 65,536 characters, two to the 16th power. The Intel 8086 shown in Figure 5 has 20 address lines, and thus an address space of 1,048,576 characters. Note that address lines and address size do not necessarily relate. If a computer has more address lines than it has bits in its address operands, some form of memory management must be used. For example, if a computer has 16-bit operands (giving a 65,536-byte directly addressable memory range) but 20 address lines (making over a million bytes of memory address possible), something must supply the extra values for four address lines which don't have any address bits in the operand to be associated with.

Intel, and most of the other vendors shown, obviously did not build a chip with address lines which could never be used. The extra four bits on the 8086 or 8088 are filled in by the IU based on a concept called memory management, or segmentation. The details of segmentation are complex, but the idea is that the instruction really doesn't need to be able to reach all of its one million memory positions at a time. Segmentation lets the programmer define an area of memory which the microprocessor is working with, called a segment, which is 65,536 characters long. The instructions only have to locate which element of the segment is being used, and if the program wants another area of memory it must change the microprocessor register which identifies what segment is being worked with.

Segments can be less than 65,536 characters long. As segments get shorter, they are more often called "pages." Page-oriented computer systems often introduce another term into what must already seem like a complex area (which it is). That term is virtual memory. Virtual memory is the amount of memory which a program can be written to assume will be available. But unlike systems which rely on "real" memory, a virtual memory system does not need to have all that memory actually available. At least, not at once.

Figure 9 shows a virtual memory system. There are two user programs, and the shaded area shows the amount of memory which each user actually uses, including the range of addresses which are expected by the program. Either program can run in the computer whose memory limit is shown with the line marked "A", since the computer is larger than the required memory. In fact, the two could be run together except for the fact that one program would have to be moved to an area where the addresses in the program would not match the address in memory (there are other ways to counteract that problem, which we'll cover later). In a virtual memory system, however, a special device called a memory management unit (MMU) will correct the conflict.

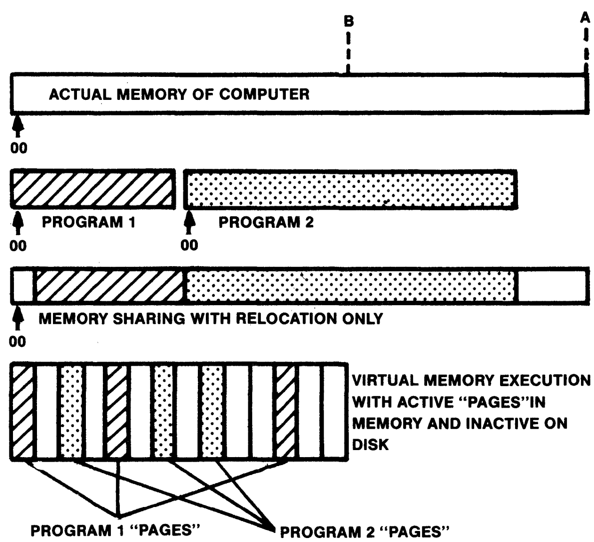


Figure 9 • virtual memory.

Let's assume that our computer has 16-bit addresses in its instructions, so it can address only 65,536 locations of memory at any one time. Instead of our segment scheme, we have a PAGE scheme in this microprocessor. In page systems, some part of the instruction address identifies a page, and the rest the location in the page. If we divide our 16 bits up to 4 bits for page identification and 12 for location within a page (sometimes called "line"), we have an instruction address range which covers 16 pages (4 bits) and 4,096 locations per page. If our MMU has its own set of registers, it can keep (for each of the two programs) the real

Microprocessor Chip Technology

memory location where each page is placed. When a program is running and an instruction references a memory location, the MMU just looks the page number up in the table to find where that page is located in real memory, then goes to the location within the page which is indicated by the "line" value.

There are easier ways to get two programs into a computer which is big enough for both, but virtual memory systems don't stop there. Referring back to Figure 9, suppose that the computer memory limit was at line "B". Now the two programs cannot both fit in memory at once, but with virtual memory they can appear to. A program can only really use three parts of memory at one time; the part where the current instruction is located, the part where the data source is found, and the part which contains the data destination. At worst case, if each of the three are in a different page, the program needs three pages to execute that instruction. The MMU permits these three pages to be anywhere in real memory. As long as we have three pages of memory for each program, and a place like a floppy disk to store the parts of the program which are not in memory at the time, we can run as many programs as we like. To help the operating system of the computer deal with pages of memory, most MMUs will generate an interrupt (we've mentioned interrupts before, and will cover them in a little more detail later) when a program wants access to a page which the computer hasn't gotten around to loading from disk storage into memory.

Issues like how the microprocessor shares its time between the programs, and what moves the pages of the program between disk and memory are outside the scope of a microprocessor chip discussion. The point is that some microprocessors are designed to work with MMUs which will provide the capability, and some are not.

Let's sum up addressing and memory. The physical memory of a computer must consist of locations for which the programs running have unique addresses, or the memory is not usable by programs. Computer instructions have address operands which can directly address a certain memory range, and internal or external memory management logic can provide various indirect means of addressing a much larger range. Memory management systems which divide real memory into large blocks, each of which is directly addressable by instructions within block boundaries but which may require special logic to switch from block to block, are called segmentation address systems. Those which have smaller blocks, called pages, and have a mechanism for translating between addresses by which the program locates the data (the instruction address) and the real address in memory are called virtual address systems.

INTERRUPTS, CONTROLS, INPUT/OUTPUT & OTHER TOPICS

As Figures 4 and 5 show, there are a number of other connections to microprocessor chips which have not been discussed. A detail description of these connections is best left to the engineers and designers, but some general comments about them may help explain some popular terms used in microcomputer literature.

Let's start by summarizing the connections to a microprocessor chip, which are generally called "lines." The lines may be grouped into data lines, address lines, and control lines. Address lines are used to inform memory (or other devices outside the microprocessor) what source or destination the chip wants to select for reading or writing. Data lines, which may share the same physical connections as address lines through multiplexing, carry the information into and out of the microprocessor chip. The collection of data lines or of address lines is sometimes referred to as the "data bus" or "address bus." The number of connections in a bus, representing the number of binary digits that can be communicated on it, is sometimes called the "bus width." Hence, you may read that a particular computer has "a data bus 16 bits wide."

The control lines are separate from the address and data lines, and are used for a number of special purposes. Several have already been discussed; the R/W line indicates whether the microprocessor is reading or writing data and the clock line(s) are used to synchronize the operation of the microprocessor with other components of the computer. There is a little less consistency in the use of the other lines, but here are some common examples:

1. I/O REQ is a line which tells all of the computer components "listening" to the address lines of the microprocessor that the address being presented on those lines identifies a device such as a terminal interface and not a memory location. This line allows the same address lines to be used to locate things in memory or to identify input/output devices.

2. Interrupt is often a collection of lines, all of which serve to tell the microprocessor chip that something needs attention now. Other parts of the computer, such as I/O devices, use the interrupt line to signal when they have information to be read by the microprocessor. We mentioned a use for interrupts in the last section, in discussing virtual memory systems. If the MMU finds that a page required by the program is not yet in memory, it generates an interrupt. Microprocessors have the ability to ignore, or mask, many of these interrupts, but most have a non-maskable interrupt line, which the microprocessor cannot ignore. This is reserved for critical situations, such as a notification from the power supply that the AC power has been lost and the computer has about a tenth of a second to clean up what it's doing.

3. Hold or wait is used to tell the microprocessor that it should stop what it is doing for a moment, not in order to do something else (like respond to an interrupt) but to permit something else within the system to perform a task which might interfere. For example, if an I/O device is writing something directly into memory (something called direct memory access, or DMA) the microprocessor could interfere if it tried to use memory at the same time.

4. Ready and acknowledge lines are used to signal that something requested has been accepted. Ready is usually used to tell the microprocessor that the

Microprocessor Chip Technology

operation it requests (read or write) can be completed by the location (or device) requested. Acknowledge lines are the opposite of a ready in that they are generated by the microprocessor to inform an outside element of the computer that a request has been understood. Hold and interrupt are examples of things typically acknowledged.

MICROPROCESSORS—WHAT'S IMPORTANT

Understanding some of the terminology and concepts of a microcomputer may be professionally rewarding but not particularly valuable in a purchase decision. Many users will find no reason to even care what microprocessor is being used in a system they purchase, and others will care for non-technical reasons only. Before asking a prospective computer vendor for the specifications on the microprocessor used in it and an explanation of the memory management system used, consider the question of the sensitivity of your application to the microprocessor itself.

Figure 10 shows a computer environment from the perspective of a computer user. Users interact directly with an application program, written by a programmer within the user's company or by a software development company. That program is written in a programming language, which was designed to work under the control of an operating system, a kind of supervisory program. That operating system was in turn designed to run on a system with a specific (or one of several types of) microprocessor. The microprocessor is thus the most distant element of the computer from the user.

From this chart, we can justify the primary rule of microprocessor selection. Unless you are writing a program in the instruction language of the microprocessor itself, you are interested in the choice of microprocessor made by a computer vendor only insofar as it affects the layers of the computer closest to you.

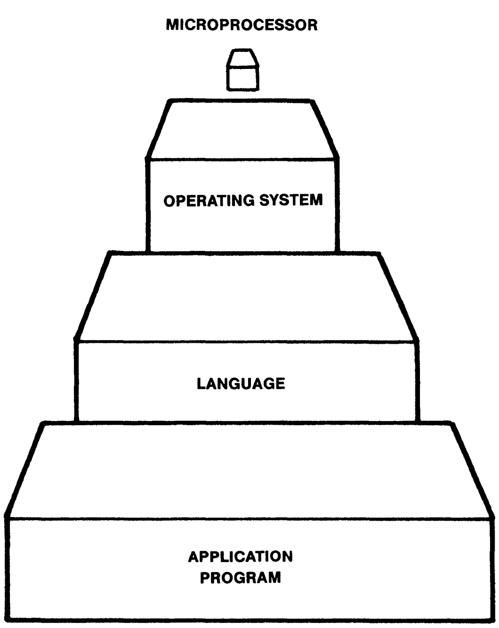


Figure 10 • a user's view of a microcomputer.

The mechanics of computer selection are covered elsewhere in this service, but understanding the selection of microprocessors requires understanding the process as a whole, so we will summarize them again here. First, application programs should be located to serve the particular requirements of the current and projected uses of the computer. Second, any programming languages which are required to support those applications should be located. If the application is already translated into computer instructions, this step is not required. Third, the operating system which is required to run the applications and programming languages must be identified, and finally the microprocessor type and computer system(s) needed to run that operating system can be considered.

This classical approach is often difficult for a corporate buyer to apply, since the purchase may be contemplated to support a more general decision support or computer literacy goal, or may be in preparation for a corporate standard of purchase which must support all applications. In this case, since the application mix will be unpredictable, it is usually best to consider systems suitable for the widest possible mix of applications. Here it is possible to apply some "indirect" microprocessor selection rules. To do so, the buyer should know what microprocessor is used in the product(s) being considered, and what are used in the major product offerings in the marketplace as well.

The first question to ask about the microprocessor is how often is it used. Not a very technical issue, to be sure, but a very important one. A microprocessor chip which is used in many products will have many skillful programmers supporting it (more have had to use it, and more often). Probably more programs will be written for it. Its popularity is an indication of market suitability, and of some history of satisfactory performance. Its vendor has probably made a lot of money selling it and is thus solvent and inclined to support it. Finally, other people with the same selection to make have obviously chosen it. While corporate purchasers should not rush off with the crowd, they should be cautious when they are on a lonely trail.

The second question about a microprocessor is upward compatibility. This is a little more technical, and relates to whether the microprocessor is a member of a family of products which provide for performance growth but retain the same basic instruction set and operating concepts. Buying the last or largest of anything leaves nowhere to go but down.

The final question to ask, and the question to be asked only if the process to this point results in multiple candidates, is the technology of the product. The evaluation of microprocessor technology is something best left to computer scientists, and if at all possible such a team should participate in any decision that gets this far. But specialists tend to select things which are interesting and topical and modern, but not necessarily practical. Managers and business professionals must participate, and should probably control

Microprocessor Chip Technology

TABLE 1: A TABLE OF MICROPROCESSORS

MODEL	DATA BUS WIDTH (bits)	INTERNAL BUS SIZE (bits)	CLOCK SPEED (MHz)	NUMBER OF INSTRUCTIONS
8080	8	16	2.6	78
8085	8	16	5.5	80
6502	8	16	4.0	56
Z80	8	16	8.0	150
Z800	8	16	25.0	183
8088	8	16	10.0	97
68008	8	32	12.5	56
16008	8	32	10.0	86
8086	16	16	10.0	97
iAPX186	16	16	10.0	150
iAPX286	16	16	10.0	163
68000	16	32	12.5	56
68010	16	32	12.5	58
16016	16	32	10.0	100
16032	16	32	10.0	100
Z-8003	16	16	10.0	110
Z-8004	16	16	10.0	110
9995	16	16	12.0	73
iAPX386	32	32	10.0	163
iAPX432	32	32	8.0	221
68020	32	32	16.0	65
Z-80000	32	32	25.0	100+
32032	32	32	10.0	86
WE32000	32	32	7.2	—
HP Focus	32	32	18.0	230
DEC MICROVAX	32	32	—	—
DG MICROEAGLE	32	32	—	—
NCR/32	32	32	13.0	179

the selection process. The primer in microprocessor technology presented in the earlier sections will help, but more specific information on the alternatives, presented in the terms already learned, will help more.

One area where users often are mislead is in the area of the "power" of instruction sets. Some microprocessor products have very flexible and sophisticated instructions, and using those instructions may improve the performance of a given business application significantly. Users may even make a selection of a microprocessor based on, for example, the ability to support floating-point calculations directly through custom instructions. Unless you intend to program directly in machine (or assembly) language, be sure that the language you select will use any special instructions before you buy a microprocessor which supports them. You might think that any language written for a computer would support all the instructions which are available on its microprocessor, but they do not.

A BUSINESSPERSON'S GUIDE TO THE MICROPROCESSOR CHIP MARKET

There are several dozen microprocessor chips on the market today, but fortunately most of them will not enter into consideration in business microcomputer selection. Those which will can be classified, and that classification is shown in Table 1. The primary grouping of microprocessors is by the width of the data bus—the only meaningful measure of "bit size." Thus the Intel 8086, the National 16032, and the Motorola 68010 are all 16-bit systems, some of the trade press notwithstanding.

Many of the chips shown in Table 1, such as the AT&T WE32000, are not normally available for commercial development but are proprietary to the manufacturer. Evaluations of products containing these special processors should not be made directly by users without specialist/engineering support, and they are included for information purposes only. Those products marked with an asterisk (*) are commonly available in commercial microcomputer products, and we'll cover each of them in more detail.

The 8-bit computers based on the 6502 (primarily the Apple II) and the Z80 (the various CP/M computer systems) dominated the market as little as two years ago. Things have now changed. A computer scientist who specialized in the development of microcomputer operating systems and other specialized software products recently admitted that "8 bits is dead." With apologies to those who love their 8-bit systems and who feel that there is no meaningful technical advantage to the 16- or 32-bit products, that's the truth. There are very few meaningful technical advantages, but the marketplace has moved to 16-bit products for now and the majority of the R&D and software writing effort is going there. The 8-bit products purchased today, and even in the near future, will still have plenty of applications and software, however. In cases where the consideration of application programs and operating systems heavily favors

Microprocessor Chip Technology

an 8-bit product, and where the investment can pay itself back in a couple years, 8-bit products may still be a good choice. The most popular 8-bit microprocessors available today are:

6502—the oldest of the current popular chips, the 6502 is known primarily for its use in the popular Apple II computer system. The 6502 is reasonably fast and inexpensive, but it has a limited instruction set and an address space of 64K bytes. There have been several efforts to extend this through memory management, most notably with the Apple III, but the newer products have not caught on. There are a lot of Apple II computers, however, and a lot of software for them. This chip is therefore likely to have the best future of the 8-bit products.

Z80—a chip developed (and a name trademarked) by Zilog, the Z80 was designed to extend the instruction set of the older Intel 8080 processor. It has a compatible superset of the 8080 instruction set, and because of that inherited the operating system developed for the 8080, Digital Research's CP/M. The combination was the hottest in the business marketplace, never approaching the total volume of the Apple II but capturing many of the complex applications and enjoying an excellent reputation among businesses and professionals who wanted the closest thing to a "real computer." The IBM Personal Computer has totally destroyed this market, and most Z80 CP/M software has been or is being converted for the PC. There is no huge installed base of Z80 systems, no single dominant market leader among its vendors, and no hope that it will be much of a factor in the future.

The 32-bit microprocessors are at the other end of the range, the newest products. Are they the wave of the future? Certainly, but the questions is "how far off is the future?" There just has not been time for a market consensus to develop on 32-bit chips—only a few are even getting built into products today. The best prospects in the 32-bit market are those which are generally compatible with their 16-bit ancestors, because programs for them can be made more readily available when (and if) they come into commercial production. Other good prospects are the chips which provide a compatible architecture to an existing minicomputer, such as DEC'S MicroVax. The ones for the far future are those with totally unique architectures, such as Intel's iAPX432.

The 68020 is the Motorola champion of 32-bit systems, and in fact is probably the premier 32-bit processor based on the fact that many people think that the whole 68000 line is 32-bit (it isn't, unless you care to accept a fairly arbitrary standard of measurement). Like the rest of the family, the 68020 is an upward-compatible enhancement of the prior products. Packed decimal instructions have been added to the product to improve its business math performance, and a math co-processor is available to improve floating-point operation. Internally, Motorola claims the 68020 to be nearly double the performance of the 68000 in 16-bit form. Since the 16-bit UNIX implementations will migrate to this with no trouble, and because of its strong virtual memory structure, the 68020 is probably the best bet 32 bits for commercial success.

Intel's iAPX386 is still a bit of a ghost, but the product is expected to be another upgrade of the segmented memory architecture of the 80286. While this provides a processor which is compatible with the earlier products, it probably pushes the segment memory management scheme a little further than most software developers will care to take it. The 386 is supposed to come with a new co-processor for match operations, an improved version of the 8087, but there is some question of exactly what kind of operations it will support. If IBM continues to go with the Intel family, the iAPX386 may find good commercial acceptance, but from the specs available it does not seem to be the hottest prospect.

National's 32032 is probably the technical champion of the 32-bit processors likely to see commercial exploitation. The product is fully compatible with the rest of the 1600 line, so mobility of code is assured. The product has a companion memory management chip with virtual memory support, as well as a floating-point chip. It's instructions are compact in comparison with other 32-bit chips, reducing program size (not much of a factor) and speeding execution by reducing the number of instruction bytes which must be pulled from memory (sometimes a significant break on execution speed). The 32032 is already offered in commercial fault-tolerant computer products, the only one of the non-proprietary chips to actually make an appearance in the marketplace.

Sixteen is best, at least in today's market. While all of the products shown in Table 1 are sound and useful from a technical perspective, the market has passed most of them by in favor of two; the Intel 8086 family and the Motorola 68000 family.

8086—the parent chip in the popular Intel family, the 8086 is now pretty much in eclipse. A version of the 8086 with an internal structure identical to its parent but with an 8-bit data bus, called 8088, was selected for the IBM Personal Computer. This shot the family to the top of the market charts, and prompted Intel to do some new engineering work to keep it there. A result of this was the 80186, a form of 8086 with a few new instructions and with a lot of circuits which were external to the old 8086 built into the processor chip instead. The result is a lower chip count and a lower computer cost, so the 80186 has largely replaced the 8086 in designs which use the 16-bit data path. A newer family member, the 80286, is even faster and has a very flexible memory management scheme. This chip is now being built into some computer systems and is generally instruction compatible with the earlier products. The biggest handicap Intel has with the series is also its biggest boost; the IBM PC. It seems that Microsoft, Inc, when they developed the

Microprocessor Chip Technology

PC-DOS operating system for IBM, used an interrupt which was reserved for use by the hardware in future developments. The use of the interrupt was written into most of the thousands of programs written later for the PC, and the newer Intel chips such as the 80186 use the same interrupt for an internal hardware feature. Until that issue is resolved satisfactorily, PC programs will require some changes to migrate upward to the new processors.

68000—a Motorola product, the 68000 actually enjoyed a brief lead over the Intel family until IBM made its decision. Unlike the 8086 computer series which uses segmented memory only (as discussed in an earlier section), the 68000 has the ability to address up to 16 million bytes of memory directly. The 68000 family is a virtual memory, page-oriented microprocessor. It also has an instruction set which is very similar to that of the popular DEC PDP-11. This combination has made the 68000 the prime target for the UNIX operating system among microprocessors. UNIX offers PC users an excellent migration path, because it is supported on computers up to the mainframe class. It's primary drawback is the relative lack of good commercial software for UNIX. Most UNIX programs are written in the "C" programming language, an uncommon vehicle for commercial software but a good language nevertheless. The 68000 comes in several models, the current champion being the newly upgraded 68010. There's even a 68000 with an 8-bit data bus, the 68008, which offers a kind of Motorola 8088 flexibility but which has not really caught on. This processor has expanded page handling capabilities and an improved scheme for pre-fetching instructions to the ALU always has something to work on. Apple Computer's latest products, the Lisa and the Macintosh, both use 68000 microprocessor chips, but at present both the user and the developer are somewhat insulated from the chip by the language and operating system structure of the machines. If you think you need a 68000 chip for some technical reason, it is highly likely that neither Lisa nor Macintosh will let you get close enough to it.

16000—National Semiconductor's family of microprocessors offers many excellent features, but has had little commercial use in microcomputers. The family has members with 8-, 16-, and 32-bit data paths, generally upward compatible throughout the range. The 16-bit chip, the 16032, has the same 16M byte address space as the 68000, and an accompanying MMU offers virtual memory support. The instruction set is considered by many to be the best on a microprocessor, and several computer manufacturers of fault-tolerant systems have based their designs on the national product. There's a UNIX operating system implementation on the 16032, and most of the popular language processors are implemented for it. Given a chance, the 1600 series could be the basis for a very powerful computer system.

SUMMING IT UP

Most users should never become involved in technical evaluations of microprocessor chips because there are too many other factors, such as languages, applications, and operating systems, which are more important and will eclipse the chip technology in an evaluation. Getting into the details of microprocessor features would normally be a consideration only when a long-range view of vendor products is being taken, or where machine-level programming is to be done by the user.

Chip technology will affect long-range suitability of products, because lack of a clear and easily supported migration strategy on the current chips will ultimately leave users stranded when better products become available. None of the current 8-bit chips have a really good migration path; they rely on high-level language compatibility rather than machine-level compatibility. Anyone who uses BASIC on several systems know how compatible high-level languages are. The 16-bit chips, on the other hand, all offer a good path for growth into 32-bit technology. IBM's use of the Intel 8086 family in the PC is probably the most significant reason to expect that series to continue to grow, but there are some indications (such as IBM's recent announcement of UNIX for the PC) that IBM might consider movement to a 68000 product in the future. IBM already has a scientific microcomputer system based on the 68000. If IBM does move convincingly to support UNIX on the PC (with full language support, conversion utilities, etc.), even without direct support of the 68000 it would weaken Intel's position.

Programming at the machine level is a good reason to consider the specific hardware. Most businesses will not find sufficient reasons to justify assembly language programming, but some special communication or device control applications may require it. There is no question that the Intel 8086 family is more difficult to program than the other 16-bit products, since the segmentation architecture seems to cause minicomputer and mainframe programmers more transition difficulties than the structures of the 68000 or 16000. Other factors may affect a programming task on microcomputers, however. Not all computers provide an assembly language (Lisa and Macintosh do not) and some systems use an internal hardware architecture which restricts some of the features of the microprocessor. Don't select a computer because of the Motorola 68000 virtual memory addressing system to find that the computer vendor did not elect to supply memory management.

The microprocessor field is changing rapidly, and most of the changes are accompanied by increases in product complexity. Very few IBM mainframe programmers even know the internal details of the software, much less of the processor itself, yet users evaluate and purchase IBM mainframes. Microcomputer buyers may find themselves lured into excessive technical detail when evaluating systems. If your application really demands attention to the microprocessor itself, that level of consideration becomes justified. In most cases, it is not. Developing a basic understanding of microprocessors is desirable for anyone who intends to use or purchase microcomputers, but that

Microprocessor Chip Technology

understanding should not be allowed to draw business professionals so close to the technology that they forget the application.

• END

Local Area Networks—a PC Perspective

■ INTRODUCTION

Recent developments in the use of coaxial cable to connect terminals or computers has stirred user interest in one of the oldest problems in data communication; the connection of locally placed computers and computer devices. Microcomputer users have a special interest in local networks as a means of sharing resources such as disk drives or printers among several systems. Much of the interest in local networks has centered on cable technology, but cable is by no means the only solution, and may not even be the best. This report examines the problems of local communication and the technologies which provide possible solutions to user problems.

LOCAL DATA COMMUNICATIONS—THE ISSUES

Studies have shown that the majority of data communication takes place between devices which are separated by distances of less than 10 miles, and many experts feel that this statistic is under-stated by the fact that many users do not regard local communication as communication at all. As the number of computer applications has grown, and with the dramatic increase in the number of personal computer systems which must access central data resources, the problems with local connection methods and their economic impacts have multiplied. Because many of these problems have no parallel in the more familiar realm of long-haul data communication, they pose a particular challenge to network managers.

The first problem in local communication is the number of connections. Many relatively small corporations have local terminal populations which reach nearly the 100 mark, and some have several hundred such devices. This large number of users creates a predictable load on the computer facility itself, but it also creates problems with the installation, operation, and maintenance of the terminal population:

• Attachment of many devices to a computer requires additional ports which are likely to be expensive and may not be available to the degree desired.

• Restrictions on the length of cable between terminal and computer may force an undesirable physical "clumping" of devices, creating an operational environment that reduces productivity.

• The number and types of cables used for connection may cause problems with cable routing, local building codes, and installation costs. In some metropolitan areas, the cost of running a cable 100 feet in an office building may easily exceed the cost of the terminal being connected.

A second problem faced in local communications is the data rate of the devices involved. Most locally connected devices are operated at relatively high data rates, often 9600 bits per second (bps) or greater. These speeds are common in long-distance communications between computers or where high-performance, high-cost terminal systems are involved. Local systems, however, will employ high data rates for terminal equipment that may well cost less than 1,000 dollars. Since this is substantially less than the cost of a pair of high-speed modems, conventional solutions for providing such speeds are unreasonably expensive.

The third problem is one of connectivity. Users of the public dial telephone system or value-added networks such as TELENET, TYMNET, or UNINET are accustomed to the ability to select a destination for a data call at the terminal either via data entry or through the use of a secondary handset. The locally connected devices which are cabled directly into the computer ports have no such selection capability unless the computer itself acts as a data switch between devices. Users who require access to multiple computer systems within the same building may have to use two different terminals or resort to a form of manual cable switch.

A more recent issue in local communication is the requirement for resource sharing among small computer systems whose normal programs and operating systems are designed for single-user, non-shared operation. Part of this desire for resource sharing is based on the cost of specialized resources such as large disk systems, but the primary motivation is the growing need to make a single, controlled, database available to more than one personal computer user. Even where economics are not considered an issue, the problems associated with maintaining parallel versions of a file in multiple personal computers may be insurmountable.

The requirements of the physical environment and the logical relationship of servers and users which resource-sharing implies can be combined to define the basic characteristics which an ideal communication facility should possess:

• The ability to connect as many as several hundred devices which may be a mixture of users and resource services to users.

• The support of device-to-device distances of up to thousands of feet.

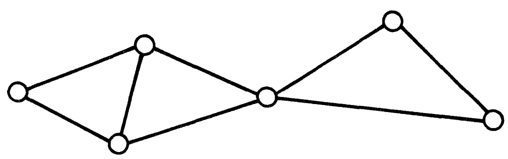
• Means of attaching host computers and other logically multioperation devices with a single physical path which supports multiple logical connections.

• The ability to support a more economical means of connection than discrete wiring dedicated to each connected device.

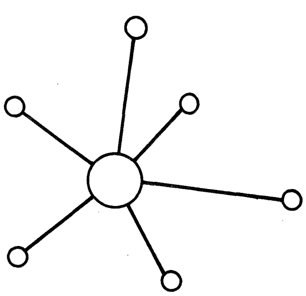
Local Area Networks—a PC Perspective

• Support for network connection control and connection services, including shared access to group resources, through a logical "connect-me-to-x" protocol rather than via fixed routes.

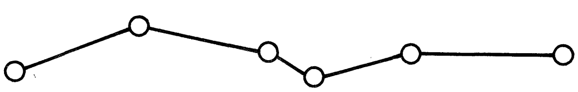
Traditional local network structures have relied primarily on switching technologies which define fixed routes between users and servers. This is because early networks were made up of a few highly intelligent (host computer) elements and many relatively dumb (terminal) devices. Configurations which consist of connected nodes, each having some intelligence and therefore some routing and connection control ability, require intelligent devices as nodes. Network structures such as the one shown in Figure 1(a), while reasonable in the long-haul world of packet switching, place an unrealistic requirement for network participation on user devices if applied to a local network. Structures shown in Figure 1 (b, c, and d) and popularly called "star" "bus" or "ring" networks can operate with a mix of devices that are each able to recognize only whether a particular message belongs to itself or to another station. This minimum intelligence can be added to the network connection mechanism easily, so that the device need not support any form of network task.



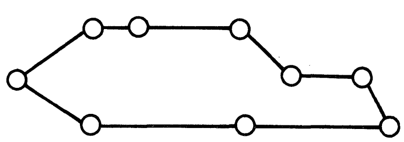
a) AN INTERCONNECTED PACKET NETWORK STUCTURE



b) A STAR OR HUB



c) A BUS



d) A RING

Figure 1 • network topologies.

The suitable LAN structures are quite different in their demands on local users, and on their support for user activities. Star networks, having a highly intelligent central core for interconnection support, may easily provide for sharing of resources because the use of the resource by other elements in the network does not involve any network users not competing for the resource. A high-speed transfer of data between the center of a "star" and one of the points loads only the two network elements involved and the non-shared path between them. In other configurations, any resource in the network which is shared or accessed by other users in other network elements will potentially affect the users whose shared transmission facilities must carry some of or all of the load. The problem is even more acute in networks where users must share a resource which is controlled by and part of one of the user's environments, and not a dedicated sharable resource to the network. An example of this is hard disk resources. If a hard disk is "owned" by a network user, other user's access to that disk will not only load the transmission paths of the network, but the user's own resources as well.

Network topology models such as "star" and "ring" may be based on the relationship between the "users" on the network. If a network represents a collection of users and only a few servers, some preferred paths between users and servers may exist and a server-central structure such as a star is ideal. Networks where the communication accesses are randomly distributed among many users will probably find that the extra cost of connecting the users to a central switching element will not be justified by the benefits. In fact, each possible way of connecting users in a local network has its own set of advantages and disadvantages, which must be considered carefully in early stages of network planning.

There are many ways to evaluate the means of connection in local area networks, and often the process begins with abstract models of network configurations. Most users will probably find that there are only a few alternatives which are practical, based on equipment already installed or readily available. For these users, the best way to start an evaluation of LAN technologies is to consider the theoretical impacts of the real connection alternatives.

DIRECT CONNECTION OF LOCAL DEVICES—THE EASIEST SOLUTION?

A directly connected network is simply one that has a server device like a computer with individual terminals, personal computers, or users connected to it through RS-232 or modem/phone line paths. This is the natural structure of many data centers, and in fact is the most common means of making local data connections.

Local-area network discussions do not normally include the option of connecting the local devices directly to the computer, but such configurations represent the star structure shown in Figure 1. In the star the computer performs a form of message switch so that devices can interconnect. Some users will find this type of solution satisfactory, however, and for a small number of connections it is to be considered cost-effective. If a device to perform port selection or switching is included in direct-connect config-

Local Area Networks—a PC Perspective

urations, the combination can serve many of the purposes of local-area networks at a reasonable cost.

Direct local connection via the normal 25-pin RS-232 interface is possible to distances of about 200 feet if the cable is specially selected and both terminal and computer are using the interface optimally. The RS-232 standard would normally limit such a connection to 50 feet, and suppliers may not be able to support an extended cable run even though they are in technical compliance with the RS-232 standard. Such direct cables can be combined with port-selection and/or port-contention devices at the computer site to provide users access to multiple computers or to allow a large terminal universe to compete for fewer computer ports.

Where greater distances are required, the user may be able to substitute another interface standard, RS-449. RS-449 will allow data exchange at distances up to 4,000 feet, and in some cases even greater range could be achieved using special cable. RS-449 is not commonly provided on terminals and may not be available on all computer ports. Adapter units are available to convert the RS-232-style interface to RS-449, and these may be an economical solution for users whose local network problems are primarily due to connection lengths. Like RS-232, RS-449 may be combined with port selection/contention devices to provide some switching and some reduction in the number of CPU ports needed to support the terminals.

Data rates on local connections are limited by the interface standard. RS-232 will provide service to 19.2K bps, while RS-449 will operate at up to 10M bps. Actual rates will depend on the distances involved, the cable characteristics, and the details of the host and terminal interface implementations. RS-449 has two sub-standards which define the signal types on critical interface circuits such as those actually used to send or receive data. Called RS-422 and RS-423, they provide a compromise between a high-performance form of the interface and the desire to retain compatibility with the older RS-232 standard, the former being a high-speed differential-drive connection and the latter being easily transformed to RS-232 for compatibility with existing equipment.

Many computer vendors have proprietary channel interfaces for the attachment of local devices, providing high data rates over reasonable connection distance ranges. Such interfaces will normally operate only with a vendor's own equipment, and use of switching or contention devices, or of the equipment of other vendors, may be impossible.

Direct connection may provide an acceptable solution to local data exchange, but it has the following disadvantages:

- Each device must be cabled directly to either the CPU or to a port selection/contention device, so cable routing and cost will remain a potential problem.
- Switching between the devices themselves is rarely possible in a direct connect mode.
- Incremememental costs for installation of new devices may be very high, especially in high-rise office buildings.

LOCAL DATA SWITCHES

Another form of the star structure is the local data switch, or data PABX. In this configuration, Figure 2, the host is replaced at the hub of the star by a switching device which can serve the terminal population and multiple host computers. Switching interaction is outside the host and therefore not loading of host facilities. This device provides terminal users with many of the same capabilities that the phone system gives to voice subscribers. Some data switches have capacities of thousands of lines and will support speeds as high as 56K bps. Most, however, operate with several hundred lines and limit speeds to 19.2K bps; neither being a serious restriction for most data users.

In its simplest form a local data switch operates as a combination of a port selector and port contention device. Each terminal has an individual line to the switch, and each CPU has a line for each available port. A terminal user may request connection to a specific CPU (port selection). If all of the ports for that CPU are in use (port contention), the terminal user may try another system or may elect to "camp on" the busy CPU and wait for a free port. Users have the

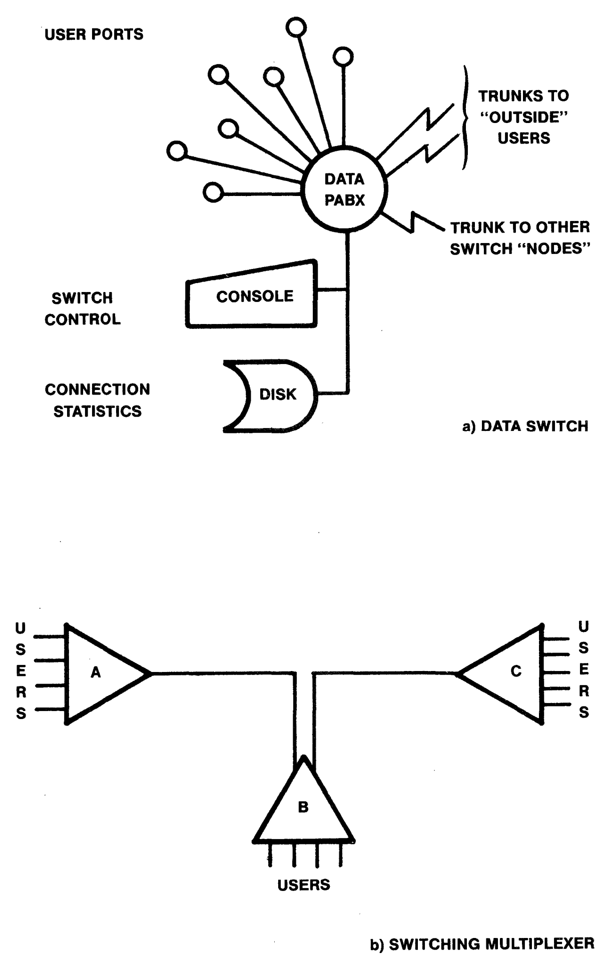


Figure 2 • data switches and multiplexers.

Local Area Networks—a PC Perspective

ability to request connection to other users, subject to compatibility of their equipment.

Data switches have another feature often found in voice PBX/CBX systems; call accounting. Because the switch is an element in each connection, it can record the destination and duration of each data call. This capability may be vital where the cost of the switching/connection facility is to be allocated according to its use. It may also provide valuable information on the loading of the computers connected to it, an aid in management planning for facilities expansion.

Some modern data switches consist of individual switching elements connected by some form of high-speed local communications path. These switch elements can be separated by several thousand feet, and each can serve as a focal point for terminals in that area. Connections between two users served by the same switch "node" can be made without access to the high-speed path between nodes, so a properly structured "node switch" can provide data users with reduced cable length and cost without impacting data performance.

Another form of data switch is the switching multiplexer. These devices combine the data concentration services of a multiplexer with port contention and switching. As shown in Figure 2, switching multiplexers look such like nodal data PBXs, but the data links between nodes can be much longer with the multiplexers since they are standard leased lines operating at rates up to 56K bps. Some switching multiplexers can be linked into multinodal networks with each node being a switching element, like a distributed data switch. On the other hand, multiplexers normally introduce more delay into the data exchanged than do data switches, and multiplexer switches are not capable of supporting as many lines per switch or handling as many simultaneous connections. The primary advantage of these systems is the ability to distribute the switching function over a wide geographical area: nodes in New York, Chicago, and Los Angeles for example.

VOICE/DATA SWITCHES

Combination voice/data switches are a new solution to data switching or local-area communications problems, but an expensive one. A voice/data switch will typically have a per-port cost of a thousand dollars, more than double that of switches handling only data. Still, voice/data combination switches have considerable advantages for some users.

Voice/data switches are sometimes called digital PABX or third generation switches because they employ a digital pulse-code technique for sending voice information, making the paths readily adaptable to data exchange. The exact method of connecting users varies from switch to switch, and the maximum number of simultaneous connections, called non-blocking maximum connections, which the switch will support is not necessarily the same as the number of user ports supported. This may not restrict a user since very few PABX systems will ever have to support as many connections as there are potential user pairs. A more serious restriction may be imposed by some switches which limit the number of data calls; or in which data calls absorb the resources of multiple voice calls, thus reducing the voice capacity of the switch.

The primary advantage of the voice/data switch is the combination of voice and data itself. Some users find that a single office position may originate a combination of voice and data calls throughout the work day that are sent to internal or external destinations. The use of a single switch and single line to handle both types of calls may save considerable equipment and cable costs.

A second benefit of the voice/data switch is the high data rates it supports. Many such switches will allow user communications at rates of 56K or 64K bps as compared to 19.2K bps for most data-only switches. This high speed is not always available to the user, however. Common asynchronous terminals are not capable of operating at such speeds. The more sophisticated synchronous devices can often employ a protocol which makes switched connections impractical or even impossible. A host is unlikely to properly respond to a terminal which suddenly "appears" on a polled line. It is unlikely that the average communications user will be able to justify a voice/data switch on the basis of data rates.

The number of ports supported by a combined switch is another benefit. Some voice/data switches will permit tens of thousands of user connections, as compared to about 5,000 maximum for data switches. Like data speeds, this may not be a significant benefit to most users because only the largest corporations could expect a user population in this range.

The final potential benefit is one of integrated call accounting. Most voice/data switches have complete call accounting packages which provide excellent charge-back information for the liquidation of switch expenses and the allocation of out-of-office telephone charges. The combined switch can provide a user with a single charge sheet for both voice and data connections. Furthermore, voice/data switches often include a feature to select the most economical circuit for a given call, taking advantage of tie lines, WATS, or other special-rate services. These features can reduce long-distance charges when the data callers are not necessarily aware of the best method of callings available at the time of the call.

Like data switches, some voice/data switches have a nodal structure. Each node is capable of handling the local switching requirements of its connected users, and may in addition connect users between nodes by routing calls over the nodal trunks. These trunk lines may be coaxial cable, parallel twisted-pair, or digital high-speed links such as T1 carrier (1.544M bps). Most systems restrict the node separation to a matter of thousands of feet, but some will support communications paths such as the AT&T T1 carrier for transcontinental separations. High-speed digital trunks have a limited value on voice connections or high-speed pulse-coded digital data since each such connection requires 56 to 64K bps, limiting a 1.544M bps trunk to only about 24 simultaneous calls.

Voice/data switching systems have recently introduced

Local Area Networks—a PC Perspective

the ability to interface to computer systems via these high-speed T1 trunks. Since a voice/data PBX can collect the potential users from any point in its connection universe and route them to the computer over this high-speed trunk, the PBX can operate almost as a front end for the computer system.

CABLE NETWORKS—THE NEWEST SOLUTION TO LOCAL DATA COMMUNICATION

In terms of user interest, cable technology has all but eclipsed the other means of providing local data communication services. Although some of the popularity of current cable systems has been due to the novelty of the concept and the interest of the trade press, there are real advantages to cable structures.

Personal computer applications have been particularly affected by the cable architecture. The interest here has been less in the communication aspects of the cable LAN, and more in the benefits of shared resources among a community of personal computer users, however. In fact, most personal computer users find the details of the operation of the communication interface between the systems to be less important to their application than the features which support the sharing of resources.

Local networks based on a form of switch represent the star architecture where terminals link to a hub, each still using a dedicated communications line. Local exchange, or switching, is accomplished by allowing the hub switch to connect the paths of the users involved. The savings in lines is limited to the elimination of dedicated connections between a given user and all of the other users with which he/she must communicate in favor of the single link to the hub. Cable networks replace this hub structure with a shared cable to which all users connect and which operates as a shared resource. The shared cable may be a single line, or bus having two end-points and a set of users connected between them, or a ring or loop path. This cable path may consist of a coaxial or CATV cable, a twisted pair of wires, a parallel-connection cable or ribbon, or even fiber optics.

One of the first issues which must be decided in cable network design is that of the way in which data will be sent on the cable. One easy method is to transmit the digital data bits directly through the cable, called "baseband" signalling. Another method, based on principles of cable television, is to send a "carrier" frequency along the cable which is modulated to contain the data in the same way in which a TV signal is modulated for the audio or video. This method, called "broadband" signalling, allows multiple uses of the same cable by separating the carrier frequencies, much like the channel separations on a CATV cable.

Topology or the form of the network is a second design issue. Cable LANs have been designed in the star, bus, and ring/loop configurations but most of the systems in use are either bus or ring structures. In a bus system, each station connects to a single long cable which may or may not be continuous from end to end. Bus systems which are made up of a continuous cable with parallel taps for users are called "broadcast" systems because all stations receive each message sent by any station on the bus. Those which are made up of a series of local connections between users are called "point-to-point" bus systems because each station sends only to the station "below" and receives only from the one "above." "Point-to point" bus systems have a repeater at the end of the cable. Ring networks, since they close on themselves, consist of a set of local paths between stations, each repeating data addressed to others down the ring toward the eventual destination.

A third cable characteristic is the means by which the information transfer is controlled. Since the cable is shared among all its stations, each station must either be granted access to the cable in a non-competing way by a master scheduler agency or all stations must compete for the cable as though it were a "party line" phone system, each having a way to detect that the facility is busy and waiting until it is free. Some cable systems separate station messages by frequency division multiplexing (FDM) or time-division multiplexing (TDM) where time slots or frequency bands are assigned for each communication. The time-slot systems have nearly disappeared in favor of either a controlled-access system operating under central control, or a multiple access system which depends on sensing the busy/free state of the cable. One implementation of the former system is sometimes called "token passing" because the right to send a message is conferred by receipt of a "token" which is passed station-to-station; and the latter is called "carrier-sense multiple access" or CSMA because each station must sense a busy condition of the circuit by testing the state of the "carrier" or line.

Most cable network suppliers agree that the exact set of concepts used in a cable network should be based on variables such as the total number of users, the data rates to be supported, and the average and peak utilization of the cable. Systems which use bus structures, baseband modulation of the cable data, and contention access are less expensive than those which use ring architectures, broadband frequency-division modulation, and token passing; but the latter set of characteristics tends to produce networks with greater performance potential and greater flexibility. Ethernet, a joint product of Digital Equipment Corp, Intel, and Xerox, is a baseband bus using CSMA, while the interactive Systems/3M offering is a broadband bus using frequency division multiplexing. IBM favors ring/loop structures for its systems, including existing offerings such as the 3600 financial system local loop and the 8100 series loop.

Where use of the cable is controlled by either contention or token management, devices normally communicate in "packets" or messages rather than in single, transparent characters. The formation of these packets and the management of the cable access method requires that the "tap" or cable interface unit be an intelligent device in itself. The IEEE (Institute of Electrical and Electronic Engineers), in its 802 committee standards effort for local networks, calls these taps "Media Access Units" or MAU's. These devices normally have a communications protocol between one another, independent of whatever protocol the user may employ to attach to the network. Cable protocols are variants on packet network protocols and are almost always

Local Area Networks—a PC Perspective

"layered" in accordance with the ISO Reference Model. Although the user protocol is converted in some sense to this internal protocol, the conversion is not complete in that some of the user protocol will pass through the cable as data. Connection of unlike device types, while a feature of interest to most cable vendors, is largely a future enhancement.

Broadband networks which use frequency separation to eliminate conflicts for use of the cable do not require packetizing or a link protocol for the cable. These systems are often called "transparent" systems because they do not introduce any packet assembly/disassembly restrictions on the user data stream.

CABLE & SWITCH STRATEGIES COMPARED

There are many more switch-based networks installed today than cable networks, and even the most optimistic estimates of strong future cable growth are unlikely to change this pattern for three to five years. Potential cable users must evaluate whether cable itself is justified in their application before they select a specific cable architecture.

Star-structured switch networks have the following advantages over cable systems:

- Very large user populations can be supported without loss of performance.
- Cost of the switch itself is normally lower than cable alternatives for networks with a larger number of ports.
- A switch can be integrated into an existing directly cabled environment easier than cable.
- Central control of the switch task allows implementation of password and other security mechanisms.
- Central accounting/billing functions are easily added and normally available from the vendor as an option.
- Network features such as logical user addresses, password protection, and closed user groups are more likely to be available.

The primary disadvantage of the switch networks is the reliance on a single central element for exchange of all user data. A failure of the switch will cause all information flow to cease unless redundant facilities are provided at additional expense. A second disadvantage is the cost of cabling each device to the switch, a significant factor if the users are widely separated or where building codes and structures require expensive installation practices.

Cable networks have the following advantages:

- The cost of a small network is much lower than with data switch networks.
- Cabling cost is almost certain to be lower than with data switches except where all users are located in the same area.
- Data transfer rates supported by cable are much higher than those available with all but the most expensive switches.
- Many cable systems are designed so that the failure of a component other than the cable itself will affect only a small number of users.
- Some cable networks permit sharing of the cable with CATV, facsimile, surveillence video, or even digital voice applications.

The primary disadvantage of cable systems is the lack of a standard. Users must expect to purchase all their media interface units from one or a small group of suppliers, and some concepts and features which become available after installation of a network may not be available for retrofit to an existing network. Further, the lack of central billing and accounting information may make cable systems unattractive to users who bill for the communications service directly or through internal accounting procedures.

Economic issues are often paramount in local network selection, since the prime goal of such networks is usually cost reduction. The cost of a network can be divided into "per-port" costs and shared facility costs. In data switches, the per-port cost is normally low because the interface to the shared facility is designed to be simple; but the cost of the shared facility (the actual switch) is high. In contrast, the cable network port costs are often higher than data switch port costs, but the central costs (the cable) are quite low.

In general, switch networks are preferred when the primary task of the network is to permit free connection among a relatively large population of devices located within several thousand yards of one another and where the devices are operating at normal communications speeds of 19.2K bps or less. Cable networks would be attractive where the primary goal of the network is to reduce cable cost; and/or where the data rates are expected to be above 19.2K bps, and/or where the same transmission medium could be used for other purposes.

CLASSIFYING CABLE ARCHITECTURES

Some of the alternative concepts of cable networks have already been defined, but a more complete list of classification points is useful in evaluating cable network design alternatives.

Topology—Networks may be a ring structure, a bus structure, a star structure or some hybrid form such as a center-connected ring. The bus and ring are the typical cable forms.

Transmission—Data may be sent directly on the cable (baseband) or used to modulate a radio or light-frequency carrier in a manner similar to a TV channel (broadband). Broadband networks may assign fixed frequency slots to communicating parties or use special modems called "frequency-agile modems" to permit users to switch slot frequencies to connect to other stations on the cable.

Control—Stations on the network may contend at random for use of the cable (CSMA), may use a form of arbitrated transfer such as station polling in the same manner as used in long-distance protocols like IBM bysync, or may pass "tokens" granting transmit permission from station to station.

Station Linking—Broadcast networks attach all stations to a

Local Area Networks—a PC Perspective

single cable and data sent from one is received by all. Point-to-point or repeater networks form the ring or bus of a series of station-to-station connections. Each station in such a string receives only from the station "above" and sends only to the station "below."

Medium of Exchange—Cable networks may use transmission-style coaxial cable, CATV cable, twisted pair or sets of twisted pairs of wire, fiber optics, or nearly any other form of electrical or optical medium.

Speed—All cable systems have a maximum internal data rate that limits the total capacity of the network. Many also have a loading limit associated with their design restricting the actual operating range to a value less than this maximum.

Data Format—Most baseband networks, since they rely on time-division use of the cable, assemble data into packets much like a public data network. This may affect network performance if the user data is not already packetized in some way, or if user messages are very short. Ethernet, for example, requires that at least 46 bytes of data be included in each "packet." This means that very short messages must be padded by the MAU out to 46 bytes, a waste of cable space. Some of the potential problems of packet assembly disassembly must be considered on this type of network. Broadband frequency-division networks, on the other hand, are normally transparent to the data structure of the users in the same way as a data PABX would be.

RINGS & BUSSES

Cable topology, or the way in which the network elements are connected, is the basic decision for cable networks in that it almost dictates the selections in the other design alternatives. The alternative selections are the ring network, where users tap onto a closed loop of cable, or the bus where they link to a single run whose ends are not connected. Figure 3 shows the various cable network structures.

Ring networks are actually structured as a series of point-to-point paths which close into a loop. Each ring tap breaks the cable, receiving data from the tap "higher" on the cable and regenerating it for the tap "lower." A form of bypass function is needed in the ring should a station fail and be unable to regenerate the message. Without the bypass, any tap failure would break the ring and make communications between some stations impossible. Rings, because they must close on themselves, will require more cable than a bus system for the same connection geography, but since all ring systems regenerate the signal with each user-to-user hop they may support greater station-to-station distances than the bus systems which employ broadcasting. Since a station in a ring must send to only one other and receive from only one other, the matching of new taps to the network is less complex and the tolerances in the receivers and transmitters are less critical. On the other hand, each new tap will require that the ring be broken, introducing the possibility of a cable fault being created during the process. Primenet, a proprietary offering of Prime Computer, is a typical example of a ring network with bypass facilities to prevent a single station failure from breaking the ring.

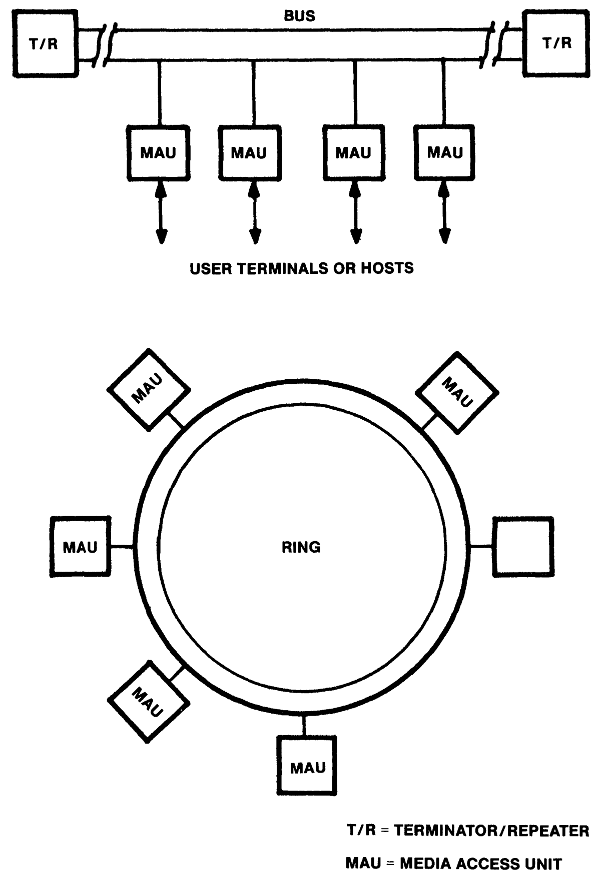


Figure 3 • ring and bus cable networks.

IBM's loop networks are examples of rings with a single controller per ring, normally called a "loop" network.

Bus networks may be constructed of individual paths in the same way as the ring systems, but this form of bus requires a repeater at the "end" of the bus to send the data back up to destinations who are attached above the sender. This downline/upline path requires either a broadband system which uses two different frequencies for the two directions or a dual-cable system. Since either of these concepts are more expensive, the most common practice is to broadcast the data onto the cable from a station, all other stations on the cable listening for their messages. The bus must be terminated at the end-points to prevent the data from "echoing" from the open end of the line back into the network. Such broadcast taps do not require that the cable be broken each time a tap is added. A simple tool can be used to attach a new cable interface unit without interrupting the cable. Some forms of continuity and quality testing can be conducted on this continuous cable to verify its operation without involving the interface units themselves. A failure of a single unit cannot cause the entire cable to fail because the interface unit serves only its own devices.

A special problem in bus networks which employ broadcasting is the propogation delay of the message. Even

Local Area Networks—a PC Perspective

though the data moves through a cable at the speed of an electrical impulse, it may take a measurable time to move from one end of a bus to the other: on the order of four to five millionths of a second. This short interval is long enough to move a complete message if cable data rates are high enough; in the million-bit-per-second range. The delay effect depends on the method used to control access to the cable, and its impact on CSMA strategies is covered in the following section.

There are many variations on the basic ring, star, and bus structures. Daisy-chaining alternate stations together with a secondary link may be used on ring networks to eliminate the problems with loss of ring continuity due to station failure. All of these structures, however, should be considered special-purpose networks for use under conditions which justify almost customized equipment, software, and installation. They are unlikely to be offered commercially at a price which can be justified by most user needs.

BROADBAND & BASEBAND NETWORKS

Most early cable networks were designed to send the data directly on the connecting medium in a digital form, called "baseband" transmission. Baseband systems can be compared with the AT&T digital service, DDS, in that they do not require modems for the communicating parties. Although even baseband systems use a form of "modulation," or a technique to convert the digital data bits into cable electrical impulses, baseband systems do not impress the data signal on a radio-frequency carrier. The interest in baseband concepts in the early networks grew out of the desire to save modem costs, at that time considerable. The cable television industry changed the economic picture by making devices for the handling of radio-frequency carriers both widely available and inexpensive. Broadband systems based on CATV components have competed favorably with the baseband systems in cost, and offer other significant advantages. Figure 4 shows a typical example of frequency usage in broadband and baseband cable systems.

Baseband systems use special modulation techniques to convert digital bits into electrical signals on the cable, a concept much like the coding of a television picture into impulses as done by the TV camera. The resulting signal can be converted back to its digital form at the other end, but two stations converting data to this baseband form would create signals which would interfere with each other if sent at the same time. Baseband systems thus must provide some means for passing control of the cable to a station so that data can be sent without the interference of other stations. Several schemes have been used, beginning with simple time-division multiplexing or "slot" concepts such as the Cambridge ring. Most modern baseband networks use either a contention system, CSMA, or a rotating-control method called token passing. CSMA networks rely on listening strategies to avoid collisions of data, while token networks circulate a "permission to send" token which a station captures in order to use the cable and which is regenerated when the message is complete.

In broadband systems, the digital signal is modulated onto a carrier frequency in the same way as a video signal is modulated onto a single TV channel frequency. The resulting combination of data and carrier can be received by another station by "tuning" to the carrier frequency. Broadband systems may use modems that can change frequencies at will to pair sender and receiver on an unused "channel" for communication. Since the sender and receiver have a special frequency, there is no need for time-division strategies such as CSMA. Some broadband networks use frequency division exclusively, while others such as WANGNET divide the cable into "bands"; some of which use fixed frequency pairings; some frequency-agile modems for switched frequency-division multiplex communications; and some of which use CSMA techniques for low-volume exchanges across large user populations.

Broadband systems also allow the users to separate from one another by using different frequencies, and allows a cable network to share a cable with television, facsimile, or other uses so long as the "channel" used for data transmission is not shared. MAU's for broadband systems may be designed to operate at a fixed frequency within the broadband network's allowed range or to move about within that range. Fixed frequency modems are less expensive but require that the sending and receiving stations have modems of the same frequency, essentially establishing a non-switched environment. Frequency-agile modems can switch frequencies to connect between any two parties on

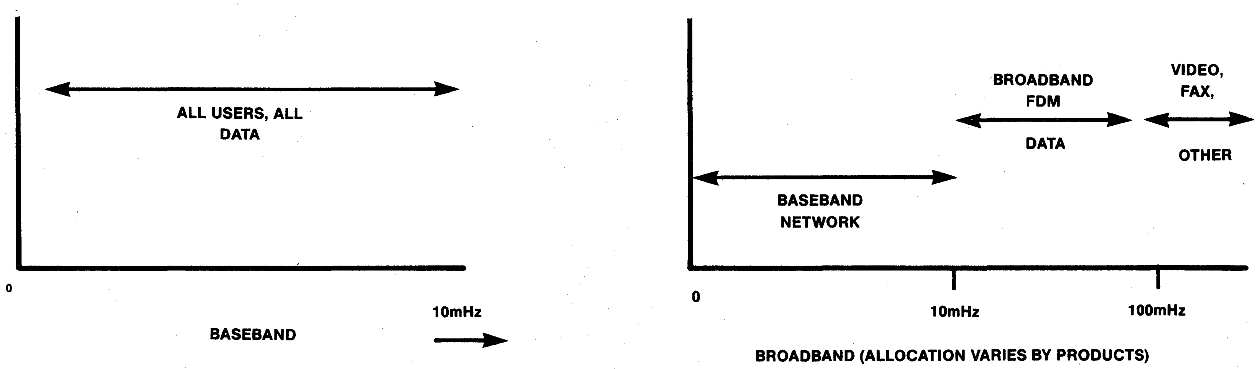


Figure 4 • broadband/baseband frequency uses.

Local Area Networks—a PC Perspective

whatever portion of the cable band happens to be available. Frequency-agile modems are much more expensive than the fixed-frequency type, and the cost difference is one of the reasons that quoted costs for broadband access taps varies so widely. Frequency-agile systems have the potential for almost incredible traffic movement. A single channel slot on a CATV cable provides 6 million cycles per second of bandwidth . . . enough for about a hundred channels at the highest data rate currently in common use, and all at the same time.

The potential traffic handling of broadband systems is considerably greater than baseband, and the baseband concept has greater growth potential than broadband. Broadband networks currently out number baseband systems, partly due to the fact that the economics of cable networks favor large installations which have high data volumes and are thus broadband targets.

CONTENTION VERSUS TOKEN—WHO CONTROLS THE FLOW?

Like the conventional long-haul networks, local networks need a way to control the access of each user to the shared facilities. Several hundred technical papers have been written describing alternative strategies for access control, but the process of commercial selection seems to have produced two principle alternatives: contention and token passing.

Early cable systems used time-division multiplexing to separate users and prevent collisions. Time slots could be visualized as "passing around" a ring network and being filled with data by the appropriate station. Such systems have much the same disadvantage as time-division multiplexing; a station has time allocated whether it is needed or not and the capacity of a network is limited by the number of slots available regardless of the chance that they are used.

Token passing (Figure 5) is a system similar to the circulating time slot. A station with data to send must first acquire a "token" which grants permission to transmit. Having done so, it removes the token and substitutes its data, regenerating the token at the end of its message. The token is produced by a station when the network is initialized and continues to circulate for so long as the network operates. Since each user regenerates the token in some way or another network should in theory operate forever on the single token in circulation. Malfunctions in the stations of the network or in the cable medium can cause the token to be damaged or destroyed. This condition will cause the network to cease to pass traffic, since no station can acquire permission to use the cable. Token networks have strategies for detecting the loss of the token and forcing a reinitialization of the network to regenerate it.

A simple alternative to token passing as a control mechanism is presented by contention systems, popularly called "carrier sense multiple access" networks, Figure 6. CSMA was popularized by its use in the University of Hawaii's ALOHA packet radio network. In this type of access control, a station simply sends data when it detects that the line is free. People who have tried this strategy at a crowded

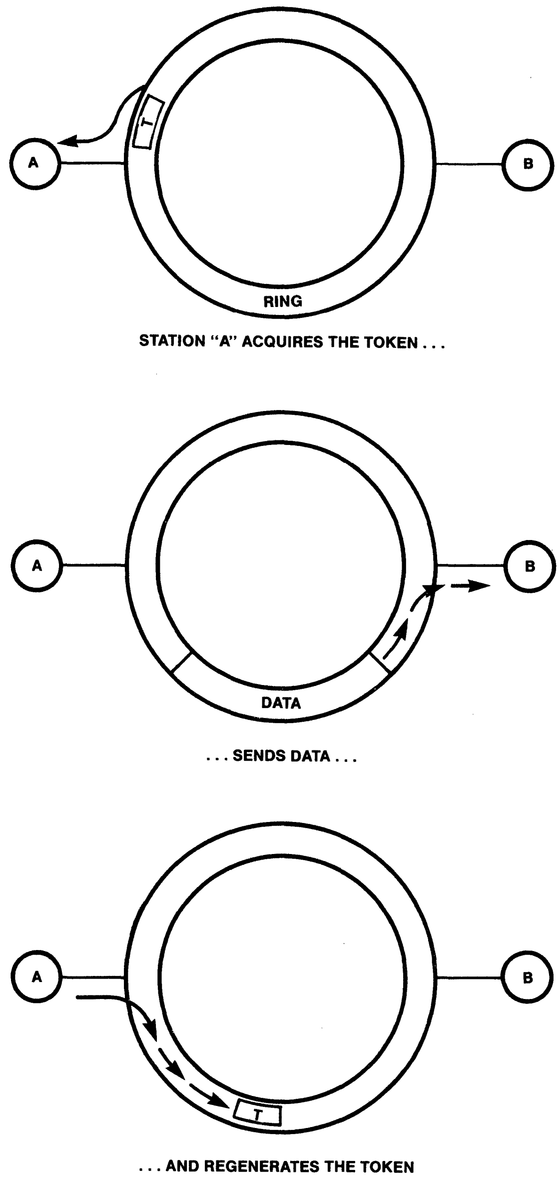


Figure 5 • an example token passing.

dinner party have already deduced its deficiencies—as the number of stations increase the chances of getting a message across reduces to zero because another station will jump in at the same moment and the data will collide. The problem is increased by the propagation delay mentioned earlier. Since both ends of the cable are some number of millionths of a second apart, a collision may occur simply because one station's message has not reached the other end of the cable. Message collision is therefore inevitable and must be dealt with in CSMA strategies.

One way to deal with it is to ignore it on the assumption that control over the number of users and messages can reduce

Local Area Networks—a PC Perspective

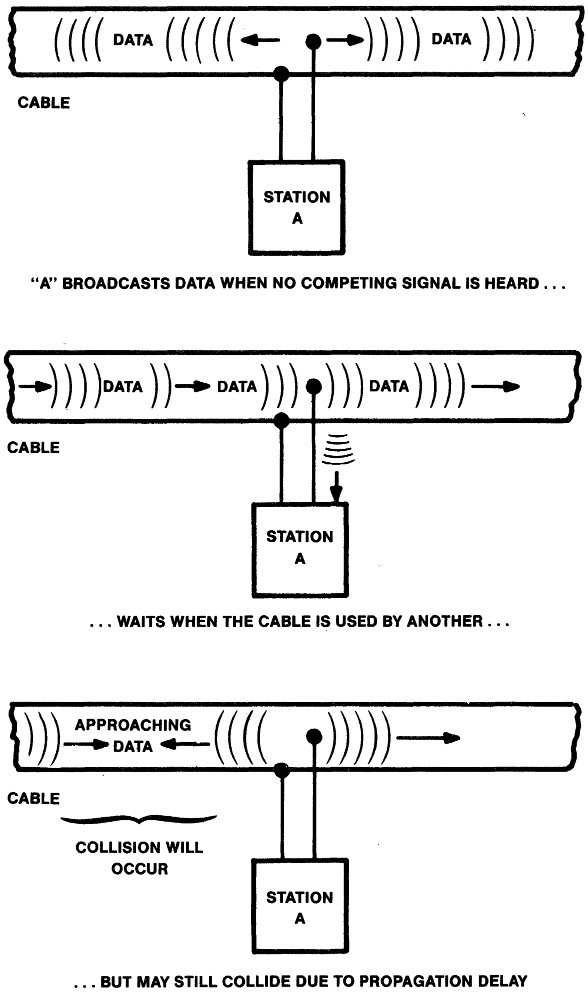


Figure 6 • CSMA.

the probability to an acceptable level. A corollary to this strategy is that the stations who have been involved in the collision can in some way try their messages again later. The "backoff" mechanisms used vary. Some systems will wait a random time interval before trying again. This randomness is needed to prevent synchronization of the competing attempts to send, resulting in deadlocking the network. Another method is based on random number generation coupled with timing measurements based on the interval of time it takes for a message to travel the entire length of the cable. This system, called "persistence," requires that a waiting station essentially "roll the dice" each time the propagation interval passes during an idle channel period. If the correct "number," called itself the "persistence," comes up the station will send its message. Otherwise the station will wait another cable delay interval and try again. Typical values of persistence are quite low, less than 10%. Networks which allow collision and depend on backoff mechanisms only are called "listen-before-talking" networks.

An obvious problem with the listen-before-talking system is that much time is wasted in sending a message into an already-existing collision. That time can be partly recaptured by having the stations listen for another message while sending their own. Actually, it is not necessary for a sender to listen for collision during the entire transmission, only for the time interval at the start of the transmission equal to the propagation delay of the cable. This is because collisions can only occur for the time period it takes for the message to reach the furthest station; after this, the normal "listen first" strategy will prevent them. Listen-while-talking strategies will allow a network to detect another station's colliding message, so these networks are called "CSMA/CD" to indicate the addition of collision detection. The CD systems are more complex in that the hardware required to detect another station's message under the station's own transmission is not trivial, particularly since all CSMA systems are broadcast systems and must also contend with differences in signal levels from nearby versus distant stations on the cable.

The propagation delay of the cable also enforces a minimum message size on CSMA networks. If the time it takes to send a message is less than the cable delay, there is a possibility that the message will be completed before the sender "hears" a collision, making the error undetectable at the cable level. CSMA/CD networks normally require that there be a minimum message size, one long enough to require over a delay interval to send. If this minimum is not imposed, there must be a higher-level message acknowledgement protocol within the network to check for errors due to undetected collisions and to request retransmission.

At low levels of data traffic, CSMA and token systems have little difference in performance. When data utilization increases, the CSMA systems experience more collisions and less free cable time and degrade in performance while the token systems wait longer for tokens and degrade as well. The difference is that the time a station must wait for a circulating token can be calculated based on the maximum message length and the number of users on the cable. This precise measurement of maximum delay has led specialists to describe token systems as "deterministic," meaning that performance can be determined mathematically. CSMA systems, on the other hand, are "probablistic." This means that the average performance of the network can be predicted, but that the specific performance at any point in time can be stated only in terms of probabilities. This deterministic/probablistic difference has caused concern among users who are reluctant to rely on a network scheme in which there is a (to be sure, very very small) chance that a station might wait hours before it could get access to the cable. In fact, the risk of very long access delay in CSMA networks is quite small unless the utilization of the cable is much higher than the recommended levels and is not much greater than the probability of a lost token or a failure of the cable itself. Most users would find the performance of either network acceptable.

Ethernet is the dominant form today of a CSMA network. It utilizes collision detection, imposes a 46-byte minimum message size, and has a binary exponential backoff strategy, which means that each time a collision is detected the

Local Area Networks—a PC Perspective

backoff time is doubled. Performance testing and mathematical modeling have shown that this combination is efficient for normal medium-scale cable networks. IBM's loop products are the most common form of token ring network.

CONNECTING THE NETWORK TO THE USER

The full potential of a cable network to connect hundreds or thousands of nodes is not necessarily easily available to the user. Cable networks, like public data networks, provide a set of services to their user community. This service set, not the potential of the architecture, must be the basis for selecting a network vendor. The MAU structure shown in Figure 7 represents the hardware attachment to the cable, but does not show the logical features provided by the software to facilitate cable use.

First, some communications environments are made up of a large number of fixed user pairings, such as those between a time-sharing terminal and a host. Cable networks in this mode offer primarily installation cost and cable routing advantages, since switching or full connectivity is not a user requirement. For networks of this type, the user would prefer that the connection map be set up once and regenerated each time the network is restarted. It is undesirable that each user must act to set up the normal connection. Such "fixed circuits" are not inherent to any particular cable network type and must be supported by the vendor in some way. In CSMA networks, fixed connections support consists of loading each MAU with the address of each of its users' partners. In fixed frequency pairing broadband systems, it consists of selecting modem frequencies properly so that paired parties operate on the same channel. Some networks downline load each fixed-connection MAU with the connection parameters at initialization time.

Where fixed pairings are not desired, the user must determine whether the exchanges are long (circuit-level support) or very short (datagram support). In either type of connection, the calling user must somehow identify the called party to the network. This requires a form a service protocol similar to that required to call into a public network or use a dial telephone. If a user wishes to connect to a fixed destination upon becoming active, some networks will allow contention for that destination's addresses and report a busy condition or "call waiting" if all the addresses are in use. Where true switching control is required, a terminal or host may have a service protocol through which connections can be requested.

Even where switching on user command is available, different levels of support can be expected. Some vendors will allow users to connect only via the cable's own address scheme, which is normally a numeric code and not easily associated with either specific users or application programs. Other vendors provide a logical address system which allows users to know others by more descriptive names such as "IBM4" or "PAYRLCLK."

Another issue is the availability of protocol conversion. This is normally available on networks in the form of speed matching (connecting a 300-bps terminal to a 9600-bps host) but rarely in the form of providing asynchronous-to-synchronous connection, or ASCII to EBCDIC interface.

These extra functions in a cable network are almost always supported by microprocessor intelligence in the MAU and often by having a "control node" which maintains user lists, assigns addresses, and loads station parameters. Many

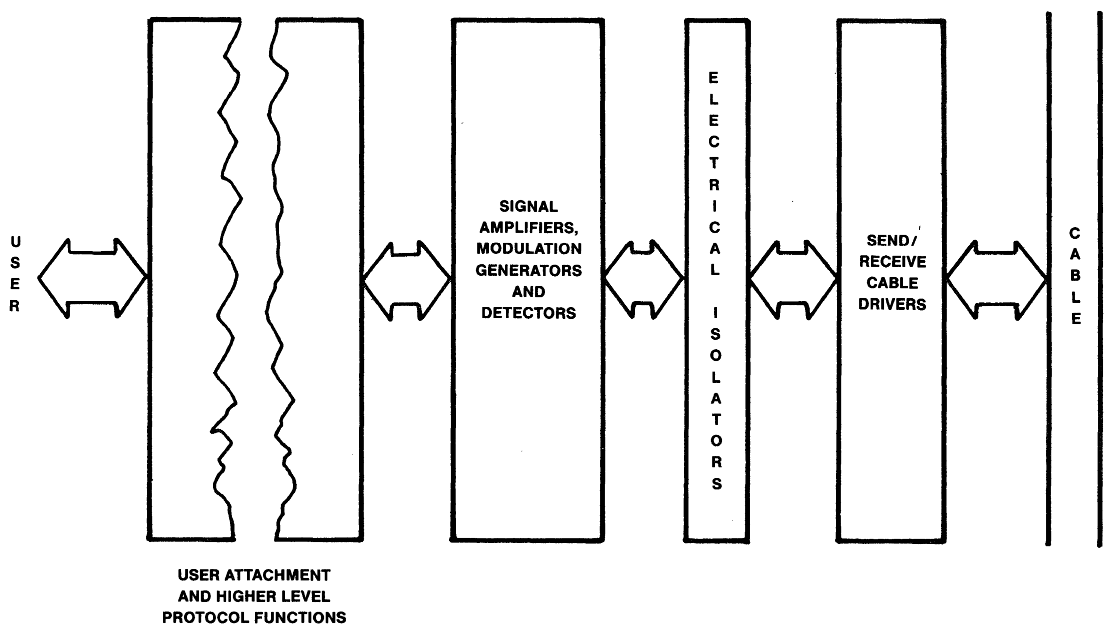


Figure 7 • media access unit.

Local Area Networks—a PC Perspective

users find these services necessary and select such a network, and yet do not consider that in doing so some of the disadvantages of central data switches have been transported into a cable environment. As the level of expected network services increases, the cable networks begin to operate like packet-switch networks and must be evaluated in those terms. Such networks are unlikely to be inexpensive and may be inferior in service, performance, and reliability to large data switches.

RESOURCE SHARING ON PERSONAL COMPUTERS USING CABLE NETWORKS

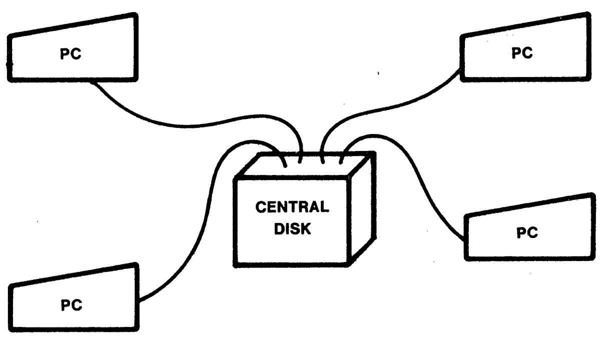
Nearly all of today's local network products designed for personal computers are based on cable technology. This fact is taken by some as an indication of the ultimate superiority of cable over other LAN technologies, but is in fact primarily a result of the fact that PC networks require a high data rate for their most important application—disk file sharing.

Users must be wary of judging personal computer LANs on communication values alone. Since the operating systems and programs most often run on PCs do not support such concepts as file sharing, and often don't even support local communication directly, the means by which the vendor chooses to mechanize these features is of primary importance in the selection of a network designed to support resource sharing. Communication issues become significant in the evaluation only when the primary issues of resource sharing features are determined to be satisfactory.

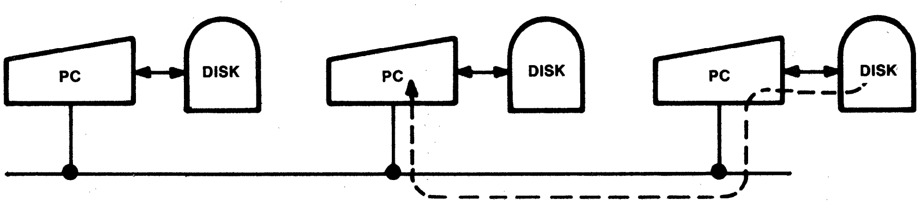
There are two structures of LANs for file sharing commonly used in PC applications, and the structure type refers less to the communication architecture than to the relationship between the resource and the sharer of the resource. If the resource is owned by a network element which has no PC and no user associated with it (a device called a "central disk server" for example), the individual sharers need access to that resource but not particularly to each other. This type of network can easily be mapped to a communication "star" technology because a star network provides exactly those characteristics. In this type of product, a central disk server provides for shared files among the connected PCs and also serves as a kind of communication switch in making communication exchanges between PCs possible. Figure 8 (a) shows a file-server network with a central "hub."

In another type of system, shown in Figure 8 (b) the disk resources are owned by one of the sharers on the network and made available to the others. This type of network may be supported by any form of communication connection architecture, because each user must connect to another in order to gain access to an external file needed for processing, but is particularly well-suited to ring or bus architectures because all systems may potentially access resources owned by any other system.

A second, and perhaps critical, problem with disk sharing local networks is the interaction between the network disk management system and the local operating systems in the



(a)
A CENTRAL NETWORK DISK



(b)
A NETWORK WITH DISTRIBUTED, SHARABLE DISKS

Figure 8 • disk sharing on an RC LAN.

Local Area Networks—a PC Perspective

attached PCs. When a file is shared, some part of the file management system must insure that two users cannot simultaneously attempt to update the same disk record. Note that it is not necessary to have the two programs access the same logical record; two adjacent customers who happen to occupy the same record on disk would cause interference if accessed at the same time by two different users because the PCs cannot write only a part of a disk record. Most programs written for execution on single-user systems like the IBM PC will not expect multiple user access and therefore will take no steps to resolve collisions between users. If the application programs themselves use file handling techniques which make collisions possible (which they all do), higher-level network software cannot prevent them.

It is possible to develop programs in many languages which will successfully resolve multiple user accesses to the same file and even to the same record. It is unlikely, however, that any application packages purchased commercially for a single-user system will employ the proper techniques. Users who plan to purchase disk sharing networks for their personal computers should not expect the networks to work properly if commercial software is expected to access shared files. Specific requirements for the sharing of files (not just the sharing of disks) should be discussed in detail with network vendors.

STANDARDS IN LOCAL NETWORKS—HOW MUCH, HOW SOON?

Cable technology has exploded in the last several years, resulting in several dozen vendors competing for what was only a 100 million dollar market in 1982, but is expected to be over 1 billion dollars by 1990. It is likely that many of the offerings of today will not survive in their present form by the end of the decade, so users must be wary of customized implementations and proprietary design. The alternative would be to adopt a standard—a definition of local network operation and equipment to do for LANs what X.25 did for public packet networks.

International standards do not happen overnight. In fact, it would not be impossible that full international agreement on a LAN standard could take five years. Since the user can hardly wait that long, and since that wait could even impact the market for the product, everyone in the LAN marketplace was eager to find an alternative solution. It came with the entry of the IEEE into the LAN standards area. Committee 802 of the IEEE was formed in 1980 to standardize local cable networks and immediately became a battleground of ring versus bus, broadband versus baseband, and CSMA versus token. The committee decided to address LANs via the ISO Reference model structure, limiting their efforts to the physical layer (level 1) and the data link layer (level 2) and persuing higher level standards through the normal standards organizations. Figure 9 shows the ISO model compared with the ETHERNET model, and 3Com's model of higher level protocols for an ETHERNET-type network.

The IEEE 802 Standards Committee was unable to define a single strategy for cable networks, a fact for which it has been unjustly criticized. There is no more hope of a single standard for local networks than there is for all of data communications. Rather than attempt to impose such a single standard and render the effort valueless, 802 proposes to provide standards for the principle variant architectures. This recognizes the fact that no single cable concept will be ideal in all communication environments. The current form of the proposal offers the following alternatives:

- datagram or circuit service.
- CSMA/CD using baseband or broadband.
- token ring or token bus.

It is probable that all of the alternatives being considered will not be fully accepted; but a CSMA/CD standard, a token bus standard, and a token ring standard will almost

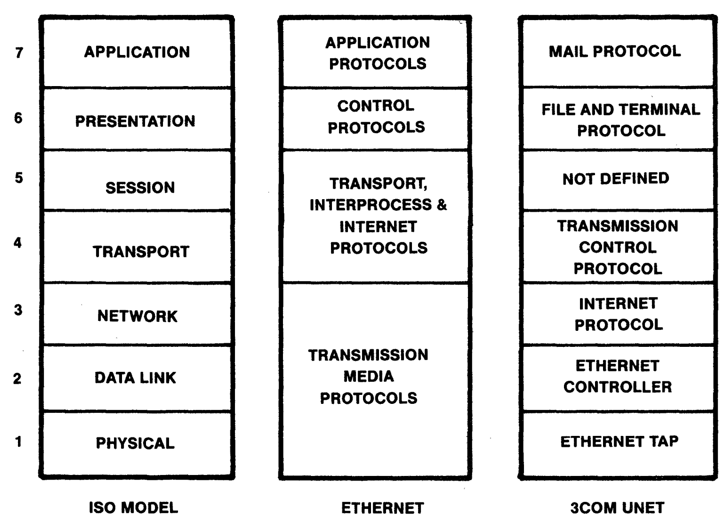


Figure 9 • protocol layering in LAN.

Local Area Networks—a PC Perspective

certainly be approved. The CSMA/CD standard differs in one particular from the ETHERNET description at this time, and it is possible that the draft may be changed, ETHERNET may be changed, or a superset agreement adopted. In general, 802 defines a superset of ETHERNET.

Other organizations are interested in LANs. The European Computer Manufacturers Association (ECMA) provided input to the 802 committee, and some changes were made in the March 1982, 802 draft as a result of the ECMA suggestions. ANSI appears ready to include the 802 standard into any future local network standard it may produce. The very short distances involved in LANs make it unlikely that international communications regulatory bodies such as the CCITT will get involved.

USER SELECTION STRATEGIES—APPLYING CAPABILITIES TO NEEDS

No network selection task can be undertaken without a clear set of user requirements. With local area networks, this is particularly important since LANs almost always carry key information within a business, information whose loss or delay could cause a disaster. The first step in selecting a local network is to study the information flow which that network is to serve. The following questions will serve as a guideline in evaluating information exchange:

• How many users will be connected? Be sure to count both the number of terminal devices and the number of host computer ports. Also be sure to project reasonable growth over the next three to five years. Don't buy a network which cannot match your growth.

• What data rates will be supported? If terminals and computers are running less than their best speed because of restrictions in the current network, don't assume that this will continue. Also, remember that most personal computer networks employ at least some disk sharing, and this demands very high data rates to be effective.

• What is the average number of characters exchanged per hour, and what is the peak volume? Be sure to add both the host and terminal traffic. In personal computer applications, this may require analyzing the number of characters of off-system data which each member of the network is likely to need. If you are considering a CSMA network, be sure to get this right because CSMA delay increases exponentially once the network becomes heavily loaded.

• What type of connections must be supported? Most users will require at least some permanent connections for host-to-printer paths. Switched connections where the host computer "dials" the number may require software changes in the host, and terminal users must be able to understand and complete any terminal switching procedure. Also, large populations of terminals in a freely switched environment almost force logical address schemes or all users will require "phone books" and spend excessive time looking up numbers or recovering from bad connections.

• Are there any incompatible connections possible among the connected users? If both asynchronous and synchronous terminals can be connected to the same network, a misdial may cause a connection between incompatible devices. This might cause a "hung port" or force manual recovery procedures, certain to be time-consuming and resented by the operating personnel. Switched systems with incompatible devices attached should have network protection against illegal connections.

• Are there any security problems with any of the users on the network? Providing free user access to the payroll system will be appreciated by some, but not likely by management.

• What are the local building codes in the area of cable routing and use? Baseband systems sometimes cause a problem with grounding because the cable is continuous. This is particularly true if the network spans several buildings. In all cases, some types of buildings are difficult to route new cables through, and some local codes will not allow certain types. Checking installation costs is also recommended; a New York cable user found that routing a cable from one floor to another in a high-rise office building cost more than a small computer system.

• Are there shared uses of the cable? CATV or other video applications can share a broadband cable, saving considerable installation cost over separate facilities.

• Is the network intended for use in a disk resource sharing environment involving personal computers? If so, are the programs which will be run to be purchased commercially or specially developed? Networks designed for disk sharing among PCs will not typically support multiple user access to the same file with commercially designed software.

Once the network needs are determined, they can be matched against the capabilities of the vendors. Selecting an architecture before looking at vendors can cut down on the number to be evaluated, but should be done only if a need clearly requires it. For example, a requirement to share the cable with surveillance TV is a legitimate reason to require broadband systems, but the assertion that the "need for datagram support requires a baseband CSMA system" is a prejudgement which should be avoided.

In reviewing vendors, look for a reasonable number of installed systems but do not expect to find five years of user experience—the architecture is not old enough. Be wary of very small firms with no capital base, no user base, and a proprietary technology. If possible, select a vendor who either implements a standard concept or shares an architecture with other vendors.

Finally, users with no data communications experience should carefully consider the use of each vendor's network concept from an operations point of view. Any network will necessarily have some performance and operational differences from a direct-connect environment, and users with no other communications experience may have to visit another installation to help visualize the effects of the new system. Do so if you have any doubts.

TRENDS IN LOCAL COMMUNICATIONS

Over half of the data exchanged travels less than 500 yards. The market for data switches as local network devices is not easily separated from other uses of such

Local Area Networks—a PC Perspective

devices, but most industry experts feel that over 100 million dollars of local networks were sold in 1981, and that the total by the end of the 1980s could near 2 billion. Cable networks seem to have captured less than half of this market at this time, and are not expected to improve their share dramatically.

One major problem with cable systems is their lack of maturity. The physical mechanisms for exchanging data on a cable are still hotly debated, and almost no progress has been made in defining higher-level protocols for cable systems except at a vendor level. These high-level protocols are just as vital for the effective use of cable networks as they were for the use of public packet networks, and their absence will continue to force users with a need for network features into the PABX camp.

Standards efforts in LANs are progressing, but the insistence that a single standard could ever be found to satisfy the diverse needs of cable users or take advantage of the wide range of transmission media will not benefit either users or vendors. Baseband CSMA systems are currently most likely to be standardized, partly because the 802 committee agrees with the main thrust of ETHERNET and partly because integrated circuits are becoming available for the 802/ETHERNET CSMA/CD protocol, making development easier and cheaper.

One area where dramatic growth can be expected is in the area of personal computer networks. The growth in the corporate use of PCs has increased the number of cases where multiple users can benefit from sharing a single disk system, printer, or communication facility. At the present time, personal computer operating systems have only the most primitive support for such applications, but newer multiuser or multitask systems, such as the PC/IX operating system recently announced for the IBM PC, may resolve some of the problems with file sharing by providing a development environment where multiple user synchronization is provided by the file manager in the operating system and sustained through the file handling logic contained in the programming languages. Electronic mail applications within an office environment where PCs are heavily used may in itself justify a network, and as the number of personal computers in office applications continues to grow, this basic communication application may be expected to follow.

In the long term, most experts agree that broadband systems offer higher data movement potential and greater flexibility. Projections on the data capacity of baseband cable over the next 10 years indicate that improvements on the order of 15% to 20% are possible, but broadband cable can expect to gain nearly 70% and fiber optics could more than double its potential capacity. High-performance systems such as Network Systems' HyperChannel have already been proved effective as an extension of the computer bus, and reduction in the cost of such systems could make them a reasonable base for connecting the multiple computers, devices, and storage media needed for automated office/distributed processing and other applications seen as critical for business growth through the 1980s.

• END

How PCs Get to Market

A Discussion of Personal Computer Distribution Channels from a Corporate Perspective

■ INTRODUCTION

The distribution channels being developed for personal computers are perhaps the most volatile and fastest growing segment in computer marketing today. The microcomputer market is less than ten years old and the methods of marketing are still in a developmental and experimental stage. A great deal of imagination is being employed by manufacturers in the attempt to find the most cost-effective methods of selling microcomputers, software, training, maintenance, and supplies to the large corporations and individual users within such corporations. When computer products were high priced and complex, as well as high-profit items, in-person sales were effective and affordable. Today with computers in this category being essentially a commodity, in-person sales calls as the primary distribution channel are simply too expensive. New approaches are added daily to the tested methods of distribution in this young market place, and the list of alternative distribution channels steadily grows longer.

Further complicating the search by manufacturers for new alternatives is the broad diversity of the marketplace. Each segment of the customer base requires a somewhat different method of distribution. The problem is also exacerbated by the many different emerging classes of microcomputers. A broad range of product offerings would require different channels of distribution. Thus, each manufacturer with a broad range of product offerings desiring to reach each subsegment of the market is required to seek many channels of distribution. Distribution channels are thus becoming specialized to address specific market segments and by the type(s) of products offered.

■ DISTRIBUTION CHANNELS

In the mid-1970s, microprocessor kits were offered for sale by mail order. This can be viewed as essentially the beginning of the retail sale of computer products. Initially the only customers for this mail order distribution were computer hobbyists who were interested in assembling, programming, and maintaining their own computers at home. This mail order distribution channel was very quickly joined by a few EDP professionals who started their own independent retail computer stores. Today, retail stores involved in the sale of personal computers and related products exceed 12,000 in the United States. Moreover, the distribution of personal computers has now expanded beyond the retail store with a number of non-direct sales channels being utilized. Therefore, in broad terms we can now classify distribution channels as direct and indirect.

Direct Distribution

Direct distribution occurs when the manufacturer comes into direct contact with the end user. Within the direct category there are further subdivisions:

- Direct sales of a company's own manufactured products to the customer through the efforts of the company's own sales organization.
- National account sales force where a selected sales group concentrates on large establishments or governmental agencies.
- Company-owned retail stores selling directly to the customer.
- Catalog offerings mailed directly to end users by the company.

The company-owned retail stores sell the owning manufacturers products directly to end users. In some instances these stores sell complementary, non-competing products (such as peripherals, media, software) from other manufacturers. These manufacturer-owned stores represent competition and thus an irritant to the other distribution channels utilized by the manufacturer and disrupt the relationship between the manufacturer and its indirect distribution outlet operators. With the exception of IBM and Tandy, it is doubtful that the maufacturers will be able to resolve the problems associated with being a major competitor to the operations they are heavily dependent upon to move their products. While the manufacturer-owned stores show a good shipment value growth over the next few years, their long-term growth is in doubt because of the real and potential conflict with the other indirect channels of distribution. With the exception of Tandy, there are relatively few manufacturer-owned stores in comparison to the independents.

Indirect Distribution

Indirect distribution channels are made up of organizations other than the manufacturer of the product, who sell to the end user. The indirect channels would include:

- Independent retail stores.
- Franchised/authorized chain retail stores.
- Distributors who act as wholesalers to supply retail outlets.
- Office equipment product dealers.
- Value added resellers.
- Merchandise marts/business centers.

Independent Retail Stores • These vendors purchase computer products from multiple manufacturers and/or wholesalers for resale to end users via walk-in storefronts with inside sales personnel. Product lines carried may be competitive or complementary. Independent retail stores

How PCs Get to Market

generally have no agreement with a manufacturer to sell their products exclusively. They often take on many lines, adding and dropping products as they deem necessary. This lack of loyalty to any given manufacturer, when tempered with an appropriate degree of concern for continuity, can lead to the user being offered a fresh selection of current, exciting products from whatever source they may arise.

Average independent retailers derive their revenue from the market segments as shown in Figure 1.

The typical independent retailer has been in business approximately 5 to 6 years and had 1983 revenue of about one million dollars with an average of four sales persons.

Discounting is very prevalent with these retailers. Their selection of suppliers is heavily weighted based upon brand recognition, image, and the product features offered. The independent retailers have not yet completely recognized the increasing requirement for sales support and training from the supplier and still place higher value on the "product" as reflected brand recognition, product image, and features. The financial considerations of inventory financing and margins are also important considerations for these independent dealers' selection of their suppliers.

The manner in which the independent dealer obtains customers is determined by the dealer's expertise in retail business areas and referrals from previous customers.

The after-market for microcomputer add-on products is a very large dollar volume business. Surprisingly, the independent dealers are not fully capitalizing on this, which points to a need to improve their customer account follow up practices. As customers, corporate users of microcomputers buying products from independent dealers are also well advised to sharpen their own awareness of the offerings in the marketplace, since their dealer cannot be depended upon for follow-up sales activity.

Franchise or Authorized Retailers • These retailers are part of a chain or network of stores where the operator pays a fee for an exclusive geographic territory for a fixed period of time. He additionally pays a franchise royalty fee based on revenue volume. The franchise/authorized dealer often sells only one manufacturers' products, although in some cases other manufacturer's products that do not compete directly with the principal brand are also carried. Generally, however, a signed commitment is made with a manufacturer to represent their product and sell a specific dollar volume annually.

The franchise retailers have tended to concentrate their operations in major population centers. Well over 90% of these retailers are located in metropolitan areas and their suburbs. They have located chiefly near main shopping areas and in large shopping malls. Nearly all are storefront operations with only a very few located in freestanding buildings. Their sales mix by market segment is quite similar to the independent retailer with the exception of having a larger share of the large company business. See Figure 2.

The franchise retailers also place extreme importance on margins in their selection of suppliers, followed closely by emphasis on product features and reputation.

These retailers also depend heavily on walk-in business which is clearly illustrated by their choice of location. As with the independent retailers, the franchise operations do not effectively draw business by advertising and direct

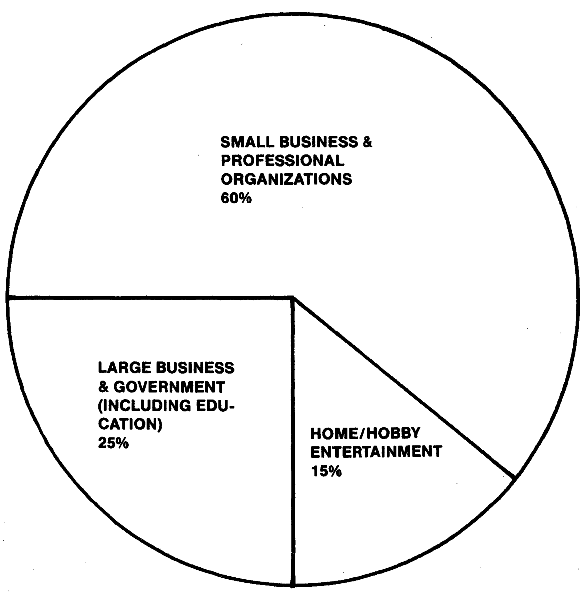


Figure 1 • approximate overall independent retailer revenue by market segment.

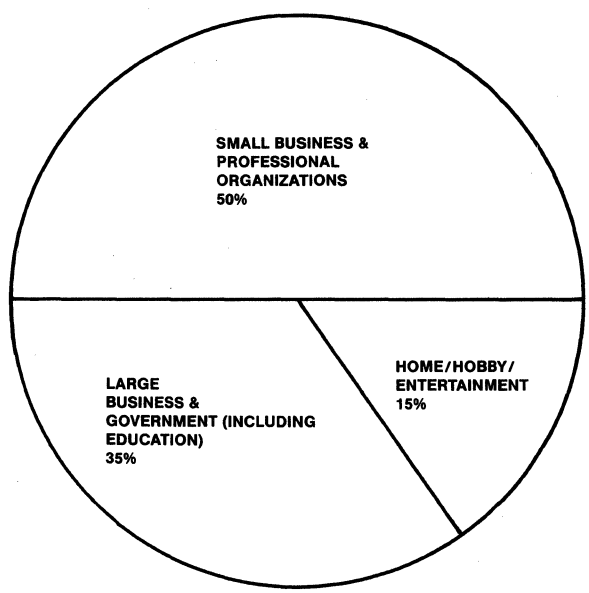


Figure 2 • approximate overall franchised/authorized retailer revenue by market segment.

How PCs Get to Market

mail/telephone marketing techniques. They do try to develop long-term relations with their customers. This makes good business sense for the retailer because these customers usually spend more on peripherals and software than they do on the original system. This also makes good sense for the corporate customer who needs a strong support structure to back up his choice of microcomputer products. Franchised retailers also are a good source of local customer referrals, which can be vital to a buyer.

Discounting practices are more widely used by the franchise dealers than by the independents with heavy discounts through advertising. The advertising budget for such a retailer, as a percent of total revenue, is typically more than twice the size of the budget for an independent.

The average franchise retailer has been in operation for about 3 to 4 years and has generated nearly $1.5 million in revenue per store in 1983. These operations average 6 sales persons.

The 3 largest franchised retailers are Computerland, Entre Computer Centers, and MicroAge. Computerland has in excess of 480 stores in the U.S. and 120 in other countries. Entre currently has approximately 145 stores and anticipates granting franchises at the rate of about 15 to 25 per month. MicroAge currently has over 90 stores.

See Table 1 for a description of major microcomputer retail outlets.

Distributors • Distributors are wholesaler-type organizations which buy products from a manufacturer and in turn market these products to dealers (the final retail outlet). The dealers are sometimes owned partly or in full by the distributor who may have many owned or franchised retail outlets.

There are basically three types of distributors:

- **Authorized distributors** are operations which sell (with their own sales staff) the products of a single manufacturer; or non-competing products from multiple manufacturers. Authorized dealers are usually given exclusive territory locations.

- **Wholesale distributors** purchase products in bulk quantity from a variety of manufacturers. Through their internal sales organization, the wholesale distributor sells products to dealers for resale to end users.

- **Retail distributors** buy in bulk quantity from the manufacturers for redistribution to a series of retail stores that may or may not be owned in part or in full by the distributor.

Distributors are secure in their position in the entire distribution chain. They have embraced the personal computer as a logical, important product for the expansion of their business beyond traditional office systems products. The distributors place a high reliance on hardware specifications and performance when choosing suppliers. They place considerably more emphasis on training and support in the selection of suppliers than do the retail operations and rely moderately on advertising to attract customers.

The average distributor has been in business nearly 15 years, and is a relatively large operation with composite average 1983 revenue of $50,000,000 and over 200 sales representatives. They are usually multisite operations having locations devoted to sales administration, shipping, and warehousing.

Distributors sell personal computers to other distribution channel operations and to the end user. Large corporations are a particularly important customer base and because of the distributors' historical role they have extremely good relationships with large company purchasing departments.

Office Equipment Dealers • The office equipment and product dealers usually sell a full line of office products including paper, office supplies, typewriters, word processors, and personal computers. The microcomputer product line is often ancillary and not the primary source of revenue. Microcomputer sales are directed to their current customer base as an extension of the service they provide to small business and professional offices. These dealers are perhaps the most conservative of all the channels involved in the distribution of personal computers. In general, they lack the experience and expertise needed to sell computers. Because of their established position in their local markets, they tend to resent suppliers establishing other distribution outlets in their area which would carry the same products they handle. They are slow to expand their lines into more complex and sophisticated products. As might be expected, these dealers are the most dependent on support from their suppliers. The typical office equipment dealer handling personal computers has the revenue profile illustrated in Figure 3.

Typically, these dealers are located in major urban areas and their adjacent suburbs, with fewer than 20% of all office equipment dealers located in rural/semi-rural locations. They are usually located on principal access roads and main shopping streets, with a small number in industrial parks. A large percentage have active expansion plans.

These dealers place considerable importance on the training and sales support provided by a vendor when making a supplier selection, and margins weigh heavily in their decision. Outside sales calls, customer referrals, and continuing business from existing customer companies constitute the chief sources of business.

Unlike most other distribution channels the office equipment dealer values a good relationship with a supplier and tends to have considerable loyalty to them.

Value Added Resellers • The value added resellers (VARs) are companies who purchase microcomputers from a manufacturer and add software value and sometimes additional hardware value. They, in turn, use their own sales force to sell directly to the customers (end users) in their own coverage area, be it local or national. These value added resellers take several forms:

How PCs Get to Market

COMPANY	ADDRESS	NUMBER OF STORES	TYPE OF OUTLET: Independent Retailer	Franchised Store	Software Only Vendor	Mass Merchandiser	Merchandise Mart	SERVICE RENDERED: Provides Training	Provides Repair Service	Provides Software Modification
Businessland	3600 Stevens Creek Blvd, San Jose, CA 95117 408-554-9300	29	•	—	—	—	—	•	•	—
Byte Industries	21130 Cabot Blvd, Hayward, CA 94545 415-787-8272	22*	—	•	—	—	—	•	•	—
CompuShack	16861 Armstrong Ave, Irvine, CA 92714 714-261-1000	7	—	•	—	—	—	•	•	—
CompuShop	1355 Glenville, Richardson, TX 07666 214-783-1252	45	•	—	—	—	—	•	•	—
Computer Factory	485 Lexington Ave, New York, NY 10017 212-687-5000	9	•	—	—	—	—	•	•	—
Computer Mart	1824 W. Maple Road, Troy, MI 48084 313-649-0910	24	—	•	—	—	—	•	•	—
The Computer Store	56 Union Ave, Sudbury, MA 01776 617-879-3700	9	•	—	—	—	—	•	•	—
ComputerCraft	1616 S. Voss, Suite 900, Houston, TX 77057 713-977-8419	33	—	—	—	—	—	•	•	•
ComputerLand (Computerland Corp)	30985 Santana St, Hayward, CA 94544 415-487-5000	509**	—	•	—	—	—	•	•	•
Computique	3211 S. Harbor Blvd, Santa Ana, CA 92704 714-549-7373	8	•	—	—	—	—	•	•	—
Control Data Business Centers (Control Data Corp)	500 W. Putnam Ave, Greenwich, CT 06830 203-622-2000	62	•	—	—	—	—	•	•	—
CPU Computer Centers	420 Rutherford Ave, Charlestown, MA 02129 617-242-3350	11	—	•	—	—	—	—	•	—
B. Dalton Booksellers	7505 Metro Blvd, Minneapolis, MN 55435 612-893-7000	800***	—	—	•	•	—	—	—	—
Digital Business Centers (Digital Equipment Corp)	146 Main St, Maynard, MA 01754 617-897-5111	52	•	—	—	—	—	•	•	—

Table 1 • top U.S. computer retail chains.

How PCs Get to Market

COMPANY	ADDRESS	NUMBER OF STORES	TYPE OF OUTLET: Independent Retailer	Franchised Store	Software Only Vendor	Mass Merchandiser	Merchandise Mart	SERVICE RENDERED: Provides Training	Provides Repair Service	Provides Software Modification
Entre Computer Centers (Entre Computer Centers, Inc)	1951 Kidwell Dr, Vienna, VA 22180 703-556-0800	151	—	•	—	—	—	•	•	•
The Genra Group	9219 Viscount Row, Dallas, TX 75247 214-688-1905	42	•	—	—	—	—	•	•	—
IBM Product Centers (IBM Corp)	National Distribution Div Retail Marketing 1600 Riveredge Parkway, Atlanta, GA 30358 404-238-2000	79	•	—	—	—	—	•	•	—
Macy's (R.H. Macy & Co, Inc)	151 W. 34th St, New York, NY 10001 212-695-4400	INA	—	—	—	•	—	—	•	—
The Math Box	1800 Rockville Pike, Rockville, MD 20852 301-984-7717	7****	•	—	—	—	—	•	•	—
MicroAge Computer Stores (MicroAge Computer Stores, Inc)	1457 W. Alameda, Tempe, AZ 85282 602-968-3168	90	—	•	—	—	—	•	•	•
Northeast Computer Stores	455 Center St, Ludlow, MA 10156 413-589-7001	7	•	—	—	—	—	•	•	•
On Line Computer Centers (On Line Microcenter, Inc)	1201 San Luis Obispo Ave, Hayward, CA 94544 415-487-2300	15	—	•	—	—	—	•	•	—
The Program Store	1945 Gallows Rd, Vienna, VA 22180 703-556-9778	20	—	•	—	—	—	—	•	—
Programs Unlimited (Programs Unlimited, Inc)	125 S. Service Rd, Jericho, NY 11753 516-997-8668	43*	—	•	—	—	—	•	•	•
Radio Shack (Tandy Corp)	1800 One Tandy Center, Fort Worth, TX 76102 817-390-3011	1114	•	—	—	—	—	•	•	—
Sears Business Systems Center (Sears Roebuck & Co)	Sears Tower, Chicago, IL 60684 312-875-2500	60	•	—	—	—	—	•	•	—
Softsel	546 N. Oak Street Inglewood, CA 90302 213-412-1700	INA	—	—	•	—	—	—	—	—
Softwaire Centres International (Softwaire Centres International)	5850 Hannum, Culver City, CA 90230 213-642-7535	53	—	•	•	—	—	—	—	—

Table 1 • top U.S. computer retail chains (con't).

How PCs Get to Market

COMPANY	ADDRESS	NUMBER OF STORES	TYPE OF OUTLET: Independent Retailer	Franchised Store	Software Only Vendor	Mass Merchandiser	Merchandise Mart	SERVICE RENDERED: Provides Training	Provides Repair Service	Provides Software Modification
Software City	1415 Queen Anne Rd, Teaneck, NJ 07666 201-833-8510	90	—	•	•	—	—	•	•	•
Software Distribution Network	17200 Ohio Dr, Aurora, CO 80017 303-752-4385	NA	—	—	•	—	—	—	—	—
Softyme, Inc	329 Bryant St, San Francisco, CA 94107 415-543-4600	NA	—	—	•	—	—	—	—	—
Team Center, Inc (Team Central)	720 29th Avenue Minneapolis, MN 55414 612-623-3850	93	—	—	—	—	•	—	—	—
Walden Books	201 High Ridge Rd, Stamford, CT 06904 203-356-7500	860***	—	—	•	•	—	—	—	—

*Since this report went to press it has come to our attention that Byte Industries has dissolved its franchise chain and will continue only as a wholesale distributor.
**Computerland also has 121 foreign locations; Programs Unlimited has 2 stores in Europe also.
***B. Dalton Booksellers and Walden Books are unsure of the number of stores with PCs. The number indicated is the total number of stores.
****The Math Box sells to Government agencies.
INA - Information not available.
NA - Not applicable because these organizations distribute software electronically.

Table 1 • top U.S. computer retail chains (con't).

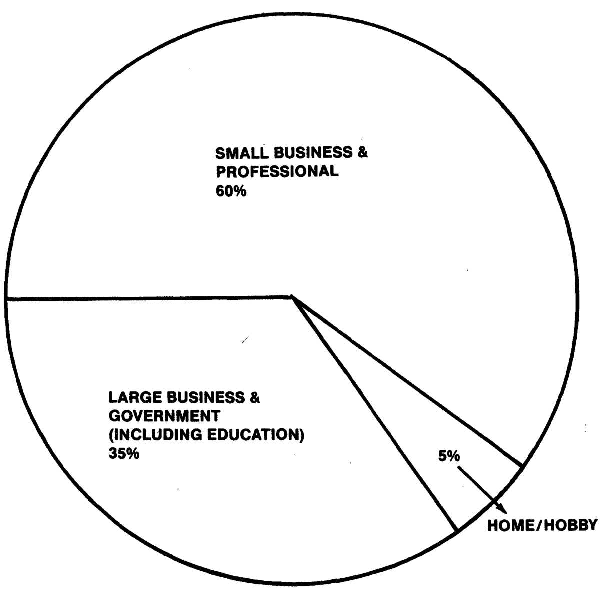


Figure 3 • office equipment dealer revenue by market segment.

- **Turnkey Systems Companies.** These companies integrate into a specific manufacturer's hardware the software designed for given vertical applications or industry markets.
- **Computer Services Companies.** The primary business of these companies is selling computing services. This increasingly involves providing personal computers/workstations at the customer's site.
- **Hardware OEMs.** These companies purchase personal computers and other hardware from different manufacturers for integration, reconfiguration, and repackaging for resale to end users.

These companies clearly focus on their target market more than most indirect channels of distribution for microcomputers. They have chosen specific industry and application areas for concentration, and because of their particular orientation, these companies place extreme importance on the particular product to which they will add value and resell. Product features dominate in the selection of a vendor. Their competition is primarily from other value added resellers rather than other types of distribution outlets for the same basic microcomputer.

The principal sources of new business for Hardware OEM-

How PCs Get to Market

VARs are the current customer base and referrals from that base. Acquisition is a popular method of expansion into new application/industry markets.

NEW TYPES OF DISTRIBUTION CHANNELS

Several new kinds of distribution channels are currently emerging for the PC marketplace. These include Merchandising Marts (sometimes called PC Supermarkets or Business Centers), Mass Merchandisers, and PC Pyramid Networks. Merchandising marts are combined showcase/retail facilities available for display, demonstration and sale of data processing products. The centers are designed for multiple supplier use which allow single-site shopping and comparison of many products. The availability of multiple supplier products represents the principal strength of this distribution channel. While some end users of microcomputers actually purchase directly at these business centers, the concept is more geared to be a wholesale operation for other distribution channel operators. A relatively small percentage of end users will go to a Merchandising Mart to purchase microcomputer products. The majority will go to the center for information purposes primarily and only secondarily may buy. The importance of this difference in the intention of the user of the center cannot be overstated. The centers must be viewed by the suppliers as a marketing promotion (showcase) activity in support of their own direct distribution and as a means of reaching the indirect distribution channel operators.

Users will not generate sufficient mart sales to justify local centers except in the largest metropolitan areas. Coupled with this is the reluctance of users to travel long distances to make use of a center. Therefore, it is questionable that these centers will ever develop into a true distribution channel in the sense they are explored in this report. However, entrepreneurs in the U.S. and Canada are investing huge amounts of money in the Merchandising Marts in anticipation that they will become important distribution channels for PCs in the 80s. Two of the most elaborate and expensive marts are scheduled to open in 1985; they are Infomart in Dallas and Boscom in Boston. Infomart is a part of the huge (7.6 million square feet) Dallas Market Center complex located near downtown Dallas; Boscom occupies approximately 1.3 million square feet in Boston's Commonwealth Pier Five near the waterfront. Two Business Marts are currently in operation, one is Chicago's Business Products Center and the other is the Bonaventure Datamart in Montreal. Several other cities have marts in the planning stage; these include Omni International in Atlanta, Techworld in Washington, DC, the California Datamart in San Francisco, the Computer Technology Center in Los Angeles, Techmart in Santa Clara, and Systemsmart in Toronto. New York is planning two Business Marts, one in Times Square called Infomart and the New York Computer Center at Fifth Avenue and 47th Street.

The Business Marts have four characteristics that distinguish them from typical PC retail outlets.

- Most exhibitors have long-term leases in the showroom; however, there are some vendors with temporary exhibits.
- The mart management staff develops themes for which special groups can conduct conferences, seminars, or trade shows. Usually the themes relate to specific businesses such as real estate, banking, manufacturing, etc.
- The marts can stage special exhibits for groups with specific interests.
- The marts will develop educational training programs to meet the buying needs of different groups of customers. For example, seminars or hardware/software demonstrations can be tailored for middle-level management groups, for clerical staff members, etc.

Mass Merchandisers are also taking advantage of the multibillion dollar microcomputer market. Several large nationwide and regional department stores, and even some furniture stores, have introduced a microcomputer department within their existing store. Others, most notably Sears, Roebuck & Co, have opened new stores devoted solely to selling microcomputer hardware and software. The Sears' stores, named Sears Business Centers, number more than 50 nationwide. Macy's, the New York-based department store, has 11 stores selling microcomputer systems on the East and West coasts and the midwest.

Another novel approach to microcomputer distribution is reminiscent of the Fuller Brush salesman. This new method involves direct, door-to-door sales of microcomputers using a network of trained salespeople. With this approach it is difficult to target a specific market, and it assumes that many people have an interest in using or purchasing a microcomputer. However, there is some activity in this type of distribution method and market research indicates that it is a viable channel. Naturally, the focus here is home/hobby/professional use rather than the type of corporate use focused on in this report.

It is still too early in the life cycle of these new distribution methods to accurately gauge their success and determine their staying power; but it is clear that the marketing savvy of the business community is playing a big role in developing new ways to market PCs.

See Table 1 for a description of major microcomputer retail outlets.

SOFTWARE DISTRIBUTION

In addition to the phenomenal increase in the use of microcomputers in large corporations in recent years, software developers have experienced the same explosive growth for their products. Market research estimates a 50% growth rate per year for micro software until 1987, providing an increase in revenues from $1 billion to as much as $5 billion. To achieve this growth rate, software publishers must utilize the most effective distribution channels for their products. Some of these channels are logical extensions of and piggyback approaches to current microcomputer system distribution channels; others are novel approaches using existing channels established for non-computer products or making innovative use of current communications technology.

How PCs Get to Market

The most common method of distributing software is to bundle it (that is, include it) in the purchase price of the microcomputer system and physically deliver it with the hardware. In many cases, operating systems, spreadsheet and word processing packages are provided by independent software companies under licensing agreement with the hardware manufacturer. Many times specific applications or productivity aids are sold with the microcomputer as a means of adding value to the packaged system. A second common method that is used by software developers is to sell the software packages through the traditional computer franchise retail store, such as Computerland, Entre, etc. It is estimated that there are more than 2,000 of these franchise operations with more than 4,000 stores. Currently, the franchised computer store is the most popular method followed by new software companies or traditional mainframe software vendors to enter the microcomputer system and software retail market. At present, a third distribution channel, the software-only stores, provide a greater depth of inventory and more comprehensive support. This type of channel is still relatively rare at present, but is expected to grow rapidly within the next 3 to 4 years. Manufacturer-owned stores present another software distribution method. As mentioned previously, Tandy/Radio Shack is by far the leader in this marketing method with 5,000+ stores. Just as with hardware sales, this type of channel for software can disrupt the relationship between the software developer and the other channel members. For this reason, it is not expected that it will become a major distribution force for software developers. For large software developers who can afford to hire and train a sales force, direct sales to large companies is another distribution alternative. However, it may be difficult for these software developers to overcome the problems created by this approach in dealing with their distributors and retailers who currently also sell to these large accounts. Two additional methods being explored by software developers are selling through traditional bookstores and by means of electronic transmission.

Many marketing professionals believe that bookstores present a logical outlet for software distribution. In fact, some large bookstores, like B. Dalton and Walden Books, have experimented with software displays and have started negotiations with software developers to obtain rights to sell their packages. However, software support will be difficult for traditional bookstores; therefore, either the orientation of the bookstore will have to change and demonstration and support facilities will have to be set up, or bookstores will sell only simple self-help software.

A final method which deserves a close watch is electronic distribution of software, called by some "teleshopping" or "telesoftware". The very nature of software lends itself to this method of distribution. This fact, together with the recent deregulation of American Telephone & Telegraph (AT&T) and its imminent entry into the micro marketplace with its vast networking capabilities, opens up the possibility of creative new methods of market penetration. (Note also that some of the newly formed "phone" companies have already announced their plans to add microcomputers to the inventory of AT&T and independent products they will offer to America's corporations.)

Currently, electronic software distribution is being test marketed in stores around the U.S. Market research estimates that this form of distribution will bring in approximately $1 billion revenue for retail stores by 1988. In order to accomplish this goal 4 important factors must be present in order to effectively deliver software to the retailers in electronic form. The factors are:

- reliable high-speed transmission over leased or switched lines at data transfer rates of at least 1200 bps.
- a fast and economical electronic receiving method to copy the delivered software at the retail store.
- a method to deliver documentation to the customer either at the time of software purchase or as an after-sale item.
- a willingness by software developers to cooperate with this new channel by supplying quality software products.

If this new method is successful, it will benefit the customer and the retailer. Customers will be able to choose from a huge inventory of software packages by utilizing a terminal in a retail outlet and receive delivery in minutes. Retailers, in addition to having satisfied customers, will be able to reduce inventory to almost zero and be assured of always being able to provide the latest version of a package.

See Table 1 for a description of major microcomputer retail outlets.

BUSINESS MIX BY DISTRIBUTION CHANNEL

Tables 2 and 3 illustrate the division of microcomputer sales in the United States market by the major channel categories. The charts show the dominance of the indirect distribution channels in the market from both a dollar volume and number-of-operations perspective. Tandy again must be viewed as an exception because of the nature of their company-owned stores.

MARKET SEGMENTS

The microcomputer market can be viewed as being comprised of four major user segments:

- Home/Hobby/Entertainment
- Professionals and Small Business Organizations
- Medium Business and Governmental Organizations
- Large Business and Governmental Organizations

MARKET SIZE

The market size for the microcomputer market is difficult to measure precisely. Many of the companies participating in this typically are small companies and some are not public. Revenue, although it equates nearly to shipment, is not published. Complicating this problem is that many of these small companies manufacture the microcomputer processor, and purchase printers and/or disk subsystems

How PCs Get to Market

	1983 $	1983 %	1987 $	1987 %
Office Equipment Dealers/Distributors/Retailers— Independent & Franchised	$2,000	36%	$ 4,000	31%
Company-owned Stores	1,000	18	3,500	27
Value Added Resellers	2,000	36	3,000	23
Direct Sales Force	500	9	2,500	19
	$5,500	100%	$13,000	100%

Table 2 • estimated U.S. personal computer market by distribution channel; cross shipments in $ millions.

	DIRECT CHANNEL	INDIRECT CHANNELS			
		Retail			
	Owned	Independent	Office Products Dealer	Distributors	Value Added Reseller
IBM	70	500	1,500	21	400
XEROX	35	—	55	25	40
APPLE	—	1,400	—	16	115
DEC	25	—	—	43	644
HP	—	—	—	—	210
TANDY	5,060	3,100	—	—	76
DG	—	80	—	5	344
WANG	—	250	—	—	83
CPT	—	—	200	—	—
LANIER	—	—	140	30	—

Table 3 • manufacturers and distribution channels used.

on an OEM basis. These sybsystems in turn are often counted again when attempts are made by researchers to measure the marketplace. This naturally distorts the statistics.

For the purpose of this report, microcomputers are defined as microprocessor-based systems ready to be plugged in and operated. A microcomputer is thus a complete system consisting of a CPU, keyboard/display, printer, storage, and operating system.

Table 4 represents the microcomputer market for the four major user segments defined above:

Home/Hobby/Entertainment • This segment of the personal computer market is by any measurement the most price competitive and volatile. The continued viability of this as a distinct market segment is in serious question since the line between the capabilities of these computers and those for business and professional use is becoming indistinct. As prices continue to fall, it becomes an absolute requirement for the various suppliers to become low-cost producers by maximizing sales volume. Additionally, the suppliers to this market segment must insure the

	1983 $	1983 %	1987 $	1987 %
Large Business and Government Organizations	$ 660	12%	$ 3,350	26%
Medium Business and Government Organizations	1,000	18	2,100	16
Professionals and Small Business Organizations	3,400	62	6,450	50
	$5,060	92	$11,900	92
Home/Hobby/Entertainment	440	8	1,100	8
	$5,500	100%	$13,000	100%

Table 4 • U.S. personal computer market by major market segments; gross shipments in $ millions.

How PCs Get to Market

availability of a number of attractive games and must attempt to place increased emphasis on more business-oriented non-game areas such as educational packages, word processing capabilities, income tax, investment, and spreadsheet applications.

The types of distribution channels to serve this segment of the market tend toward large national chains, mail order catalogs, and other types of mass distributors who are experienced in and comfortable with marketing high-volume, standard products in a price-competitive market environment. This distribution approach implies a minimum of support and to compensate for this additional emphasis is being placed on self-teaching books and software packages.

Most of the supporting products, games, software packages, publications, etc do not originate with the manufacturer of these computers but are supplied by independent third parties. These ancillary suppliers tend to concentrate on the market leaders, thus each manufacturer must try to induce as many organizations as possible to support his system and even a suspicion of faltering can be devastating. When a particular manufacturer loses market share or releases a bad earnings report, the great fear is that these independent suppliers will abandon the product in favor of a more successful and popular system, thus hastening and, in fact, sometimes forcing the demise of a product solely based upon the panic reaction.

Small Professional & Business Organizations • This segment of the personal computer market is comprised of professionals or small business establishments, who require a business oriented computer. The computer is often the central processor of the business and perhaps the only processor of the business. For this segment excellent maintenance and software support is necessary and a variety of generalized as well as very specialized business applications packages are required. There is both a requirement and an opportunity to provide additional peripherals which add to the overall system price—for example, correspondence-quality printer, hard fixed disk storage, and reasonably sophisticated word processing software.

While all the personal computer market segments are price sensitive, this market is less so than the home/hobby/entertainment segment because excellent service and support are so necessary that the small business will pay a premium price for it. If accounts receivable are stored on a disk drive that does not work, the business operator cannot afford to ship it back to the manufacturer for repair.

Acceptable channels of distribution are far more varied for this market than for the other three segments, and clear-cut distribution patterns have not yet emerged. It is likely that the distribution channels which eventually dominate will include high-quality computer retail stores (offering support, service, and a variety of peripherals) and value added systems houses which specialize in particular industries (vertical marketing) and applications and provide a complete and tailored package for particular niche businesses. The value added marketers will probably take several forms. These will range from the small local marketer/systems house, to store front operators specializing in particular niche markets, to national and even worldwide organizations. These latter may well operate under the auspices of a major accounting firm, financial service firm, or even a major non-computer manufacturer. For example, a drug manufacturer offering a packaged service to physicians or pharmacists; or an animal feed producer offering a specialized service for farmers.

At present, communications capability is not of major importance to this segment since there usually is no other processor with which to communicate. Many forecasters believe that in the future the use of external databases will become increasingly important and that electronic mail will also become attractive even to very small users.

Medium Business Segment • The medium business market is made up of establishments ranging from about $10 million in sales up to about $1 billion in sales. This size of company probably has one or perhaps several computers in use, and for the larger companies there is probably a centralized data processing complex with computers or intelligent terminals in remote locations. The key factor here is that communication capability and some degree of compatibility with the mainframes in use in the company is required. It is also necessary for the microcomputer to access data stored in corporate files and to interact with office automation equipment and other personal computers.

The medium business segment is the most complex segment as the establishments range from small companies to the Fortune 2000 customer class.

The smaller of the medium organizations are well served by the independent dealers and authorized/franchised dealers. To some extent the office equipment product dealers can also serve this market. Perhaps the most effective channel of distribution is the value added reseller (VAR). The VAR's knowledge of the vertical market served, representation in the customer's geographic area, and a direct sales force have made them well suited for the medium-sized organizations. It would appear that vertical marketing is the key to serving these organizations.

As the customer's organization increases in size and becomes nationwide or worldwide in scope, the VARs face the problem of adequate coverage to match that of the customer they are covering. Often the VARs cannot give adequate sales and service coverage to the upper edge of this category.

These large establishments are the traditional marketplace of the mainframe and minicomputer vendors who are attempting to maintain account control by emphasizing integrated product lines, local area networks, and compatibility with the major mainframe vendor used by the customer.

In conclusion, the medium business segment presents the most complex problems of channels of distribution and conversely offers excellent opportunities for the manufacturers of microcomputers. Proper definition of the niche(s)

How PCs Get to Market

being served by a retailer can easily result in gain of market share.

Large Organization Segment • The large organization market consists of the Fortune top 500 manufacturers and other giant enterprises such as banks, financial and service companies, government agencies, universities, etc. In the United States, these organizations account for about 40% of all data processing dollars spent by users and probably 30% to 35% of all expenditures outside of the United States.

In many ways these organizations comprise a market that is quite similar to the upper two-thirds of the medium organization market. The qualitative differences are in 1) the domination of this market worldwide by IBM, in 2) the trend toward full-blown corporate-wide networking, and in 3) the importance of IBM's System Network Architecture (SNA). While IBM attempts to exert strong account control over all of its customers, SNA networking provides much more leverage in this segment. SNA will almost certainly become the de facto networking standard of the future. Totally integrated data processing systems are required by this market and integration here must accommodate SNA. The requirement for excellent local support is present but may decline in this market because of the growing trend within large organizations to take control of their own in-house personal computer usage. The proliferation of personal computers has led to a realization that some standards must be established to avoid chaos. Advisory committees or departments are being established which offer advice, assistance, and often discount prices through quantity purchases to employees or other departments that wish to purchase or use a personal computer. These organizations are beginning to take over some of the burden of support and this trend can only increase in the future.

The large organization segment is more readily reached by the various microcomputers' manufacturer's direct sales forces. The manufacturer is already present with mainframe and various auxiliary equipment. It is only natural to market microcomputers to that customer base.

Often neither the authorized microcomputer dealer for a given mainframe manufacturer nor the corresponding independent dealer are skilled in penetrating and marketing to the large organization. The office equipment dealer also is placed in the same position. That leaves the distributor and value added reseller the task of marketing to the large organization. The VARs often find the small- and medium-sized organization a better target since they are offering a complete system (hardware and vertical-market-oriented software). The distributor on the other hand has sufficient financial resources and sales personnel to attack this segment. Thus, it would presently appear that the direct sales force and distributor are better positioned that the other channels of distribution to serve the market segment.

CYCLICAL & SEASONAL TRENDS

Economic cycles affect market segments in different ways. The small business user is the first to react to a poor economic climate and bookings and order backlogs begin to decline almost with the first signs of a downturn. At the other end of the spectrum, systems sold to large organizations are usually the last to be affected. In general, the small user does not have the capital resources to continue with high spending in bleak times while the larger user may already have invested such substantial amounts in planning and preparing for systems that they will usually not postpone a scheduled implementation until the economic situation has deteriorated severely.

While the personal computer market is still too new to have a clearly established track record, it can reasonably be assumed that it is and will continue to be vulnerable to poor economic conditions, notwithstanding its high growth during the recession, in its infancy, of the early 1980s. This growth occurred during the beginning stages of a new market for an extremely useful product. In such a market, strong demand will be found even in a poor economic climate. However, in the future, it is likely that the home market, small business market, and some substantial portion of the medium business market will reflect the traditional cyclical pattern of the industry. Thus, all microcomputer manufacturers and their distribution channels can expect to be highly cyclical in revenue and earnings and a major strategic goal of each should be to establish a conservative financial situation designed to survive without catastrophic losses during periods of decline and to be able to respond with vigor during recoveries.

There is little evidence to indicate that traditional computer markets are greatly influenced by seasonality. There is a slight fall of business during the summer caused by the reduced pace of business during vacation seasons. Tax deadlines also have a slight effect. The apparent seasonality of many computer suppliers stems more from the scramble to include (or exclude) bookings from one financial period into another than from true seasonal variations in the market. This may well not be the case with personal computers. Sales of consumer products are heavily weighted by the Christmas season and the personal computer has many of the characteristics of a consumer product. A general downturn in factory orders after Thanksgiving 1983 seems to have surprised many forecasters and manufacturers. Much of this was attributed to product obsolescence and attractive new product announcements by specific manufacturers. Although actual market statistics are too new for accurate trend analysis, it may well be that pronounced seasonal trends will emerge which will confound microcomputer vendors and manufacturers who are not aware of them.

CORPORATE BUYER CONSIDERATIONS

For the business buyer of personal computers it is important to approach the selection of the most appropriate channel of distribution for the company in a planned orderly manner. Many companies are now establishing committees and/or individuals to act as consultants for departments with budgets to purchase personal computers. A similar situation developed early in the marketing of minicomputers. At first, departments would use their own purchasing authority to buy a minicomputer for departmental use in a completely uncontrolled manner. Large companies could not even estimate how many minicom-

How PCs Get to Market

puters were in use throughout their companies. As incompatibility problems began to plague the companies, they started to centralize the decision making and bring the situation under control. The problem has been compounded in personal computer acquisitions because the low cost of the equipment has dramatically increased the number of individuals within companies with the authority to purchase personal computers. The same compatibility, database and programming problems have begun to arise. Hence, the move to a more controlled acquisition procedure in many companies is evident today. The decision as to whether a firm is better served by a rigid policy, a highly centralized versus a consulting advisory function, or something in between is a function of corporate policy/culture and management philosophy.

No matter what control mechanism is employed there are a number of items which should be considered in the purchase of microcomputers for corporate use relating to the channel of distribution to be used. These items should be matched against company structure, location(s), plans for personal computer expansion, and future networking plans. Once this is completed, the distribution channel which best matches the particular company's requirement should be reasonably apparent.

Table 5 is designed as a guide in channel selection for corporate users of microcomputers. The items which need to be considered are shown on the left. The various distribution channels are the top column headings. The numbers shown represent the degree of emphasis/importance the distribution channel places on each of the evaluation items. The chart can be used in a trade-off evaluation. The buyer should weight the rating for the channel based on his own priority for the evaluation items. Such a procedure will allow the buyer to select the channel(s) with the highest scores on the items of importance in his own selection criteria.

The items suggested for evaluation are:

- Product delivery
- Availability of demonstration units
- Customer training
- Service/support local to end users
- Availability of related add-on products
- Hardware evaluation practices
- Software evaluation practices
- Availability of vertical applications
- Software demonstration packages
- Documentation—hardware and software

	INDEPENDENT RETAIL STORES	FRANCHISE/ AUTHORIZED RETAILERS	OFFICE EQUIPMENT DEALERS	DISTRIBUTORS	VALUE ADDED RESELLERS	MANUFACTURER DIRECT SALES
PRODUCT DELIVERY	3	4	3	3	3	4
DEMONSTRATION UNITS	3	4	3	3	2	4
CUSTOMER TRAINING	2	3	4	2	2	4
LOCAL SERVICE & SUPPORT	2	3	3	3	3	3
ADD-ON PRODUCT LINE	3	4	2	3	3	4
HARDWARE EVALUATION	2	4	2	1	4	3
SOFTWARE EVALUATION	2	4	1	1	4	3
VERTICAL APPLICATIONS	1	1	1	1	4	2
SOFTWARE DEMO	2	3	1	1	4	2
DOCUMENTATION	3	3	2	2	3	3
PLANNING & CONSULTATION	1	2	1	1	3	3
PARTS STOCKING	2	3	3	3	2	4
MANUFACTURER RELATIONSHIP/SUPPORT	3	4	4	4	4	N/A
DISCOUNTING	3	4	3	3	1	1
STAFF QUALITY	1	2	2	1	4	3
MULTILINE PRODUCTS	2	2	4	4	1	1

1 = Least emphasis
4 = Most emphasis
N/A = Not Applicable

Table 5 • channel selection criteria.

How PCs Get to Market

- Planning and consultation availability
- Parts stocking/availability
- Relationship between product manufacturers and distribution channel
- Discounting
- Quality of distribution channel staff and representatives
- Multiline product availability

While the individual operations within the distribution channels can vary significantly, some generalization can be offered concerning the emphasis placed on these areas which should enter into the evaluation.

Distribution of microcomputers is the cornerstone of the microcomputer industry. From the vendors' point of view, channels must be selected and developed which will most effectively service their particular target markets. From the channel operator point of view, the vendors must be selected with the product and support structure to assist in serving their particular market niche. From the buyer point of view, they must carefully select, mix and match the various distribution channels available to them to most closely fit the corporate plans for the use of personal computers. These three elements in the chain have not been fully developed nor are the techniques for doing so in place. However, as the industry matures and reaches its full potential the distribution of microcomputers will have to take into consideration the best interest of all three principals—the vendor, the distribution channel and the buyer.

• END

Data Transmission Alternatives: LAN or PBX?

An Analysis of the Tradeoffs of PBXs & LANs for Local Data Transmission

■ INTRODUCTION

The trend having the most profound impact on the data processing industry today is the explosive growth in the demand for personal computers. Growth in this market has dramatically exceeded even the most optimistic projections made only a few years ago. For the next year, the primary constraint on the expansion of these systems will be the availability of the integrated circuits used to build them. While there were widespread debates as recently as 1982 regarding the nature of the office workstation of the future, it is now widely acknowledged that the personal computer will be an important element in the development of advanced office systems.

Although the personal computer has begun to play an important role in automating the business office, it has also created a unique set of problems. While providing individualized computing to each white-collar worker is a first step toward improving business productivity, it is also essential that users have the ability to exchange information among their workstations. It is on this latter issue where the current situation is far from optimal. Personal computers have significantly altered the needs of local data communications, but networking hardware and software to serve these needs is still in an early stage of development.

Personal computers have influenced both the volume of data being transmitted and the types of communication services that are required. That is, by increasing the amount of information that is handled electronically, PCs have expanded the need for data transmission facilities. Also, the procurement process for PCs is often more decentralized than that for traditional computer systems, which has tended to increase the variety of different manufacturers at a given location, leading to communication compatibility problems. Finally, the data networking architectures that were developed for centralized mainframe environments (e.g., SNA) or for minicomputers, are not typically well suited for the needs of personal computers.

To fully exploit the capabilities of PCs will require that information be easily shared among them. With this incentive, the following discussion will examine two different types of networking products that can serve these communication needs. Both Local Area Networks (LANs) and Private Branch Exchanges (PBXs) can enhance the effectiveness of PCs by allowing data to be shared among them or by interconnecting them with other data processing equipment.

■ BACKGROUND

Among the more widely acknowledged technological trends of the last decade has been the convergence of data processing and communications. Dramatic evidence of this trend can be seen in a wide array of new computer networking products, and even a linguistic impact is apparent in the emergence of terms such as "compunications" and "information management." The consolidation of different technologies has been particularly important in the business office, where office equipment joins data processing and communications to form the three cornerstones of future integrated systems. Using descriptions such as office automation, office integration, and office-of-the-future, companies that have served traditional white-collar markets are now creating new markets by offering products that cut across the conventional boundaries. Local Area Networks (LANs) are a prime example of a new class of products to emerge from this melding of technologies.

While all major vendors in the office market recognize the importance of moving from more specialized conventional products into integrated systems, they have different visions of how the transition is likely to occur. Mainframe and minicomputer vendors view office automation evolving from conventional data processing systems as an extension of the trend toward distributed processing. Alternatively, PBX vendors see their digital switching products, wired to each white collar worker's desk, as being natural vehicles for interconnecting the diverse array of office functions into an integrated system. As the third element of the triad, office equipment vendors focus on the individual workstations, with new networks (i.e., Local Area Networks, LANs) being developed to connect and integrate these previously autonomous products. This diversity of perspectives on the future of the white-collar workplace has led to many debates and occasional confusion regarding the direction that future technology will lead.

Although there are many different views on the structure of future business information systems, there is general agreement on at least one issue. All agree that the network will play an increasingly important role. That is, as separate product categories merge into integrated systems, more attention will be focused on the technologies for tying together the various system elements. With this foundation, the goal of this report will be to look at 2 different types of networks, LANs and PBXs, both of which could play a major role in feature information management systems. Both the present capabilities and future potential of these networking alternatives will be evaluated, and their advantages and disadvantages in different operating environments will be compared.

Definitions

Before beginning a discussion of LAN and PBX products,

Data Transmission Alternatives: LAN or PBX?

it is important to first define what is implied by each of these terms. Since there is general agreement on what constitutes a PBX, most of the following comments will be directed at establishing a precise definition of a LAN. PBX (or Private Branch Exchange) is an acronym used to describe a class of switching products that have been designed to serve the needs of voice communications. The basic architecture uses twisted pair wiring to connect all attached devices to a centralized switching location. While PBXs have been enhanced to offer data transmission, voice communication is still their primary function. PBX architectures and capabilities reflect this voice origin.

Unfortunately, there is not yet a generally accepted LAN definition, since different vendors have stretched the meaning of the term to suit their own promotional purposes. That is, to the extent that the promise and the capabilities of LANs are perceived by many in a positive light, companies have been quick to make use of this label, attempting to benefit from the association of their product with generic LAN capabilities. In addition, the differing perspectives of data processing, office equipment, and communications vendors have also contributed to divergent interpretations of the LAN label.

In defining a LAN, both functional and technical issues will be considered. From a functional perspective, it is generally agreed that the term LAN implies a communications capability over a "local" area, which certainly encompasses a large building and is typically understood to cover multibuilding complexes. While the term has occasionally been used loosely to include larger metropolitan-type networks, including Digital Termination Service (DTS), we will restrict the current definition to include only private networks. This implies that most LANs will be limited to an organization's contiguous physical facilities, single or multibuilding, and do not include common carrier transmission links.

Bandwidth capabilities have occasionally been used to distinguish LANs from PBXs, since most LANs are capable of providing greater information capacities to their attached devices. However, there are also data networking products that can be classified as LANs, but which offer access speeds of only 19.2K bps. This is significantly less than the 64K bps common in new PBX products, so bandwidth is not always a useful distinguishing feature.

Another functional characteristic of LANs is their present focus on data and text transmission. While the integration of voice and video is also desirable, and is being accomplished to a limited degree in some systems, data and text communications are still the primary applications. We will use this nonvoice focus as an important feature that distinguishes LANs from PBXs. That is, a PBX is a system whose design reflects the needs of voice communications, while LANs are designed to meet the needs of data communications. Note that this implies that LANs and PBXs are 2 mutually distinct categories, which differs from the interpretation of some observers. Thus, this report will not adopt the definition of those PBX vendors that classify their products as LANs. While such an interpretation is justified to the extent that PBXs and LANs can serve similar data communication applications, the origins and primary functions of these 2 types of networks are clearly distinct. The boundary between them is certain to become more blurred in the future, but present products are not difficult to place in one category or the other.

A product with similar functions that fits neither the PBX nor LAN definition proposed thus far is the data PBX. These systems use twisted pair wiring to provide relatively low-speed interconnection between terminals and computers. Architecturally they resemble a PBX but have functional capabilities similar to LANs. For this reason, this report will treat data PBXs as a separate product category. General use of the term PBX will refer only to systems used primarily for voice, and any discussion of data PBXs will be clearly distinguished.

As another point of definition for LANs, we will use the International Standards Organization (ISO) reference model for Open Systems Interconnection (OSI). From this perspective, a LAN is required to provide the services of the lower 2 levels of the ISO model; physical and data link (more on the ISO model in a later section). This is contrary to the implication of those who refer to the higher levels of communication and application software as LANs. While computer networks that allow shared files and applications must have higher levels of software to exploit the capabilities of LANs, this software alone is not a LAN. Such software can typically work over geographical area significantly larger than those considered likely for most "local" networks. Thus, while higher levels of software are certainly important in networking applications, it is the lower 2 levels that are relevant when comparing PBXs and LANs. Note that our interpretation of this issue agrees with the work of the Institute of Electrical and Electronic Engineers (IEEE) 802 committee that is working on LAN standards.

As 2 final criteria for LANs, we will exclude from our definition those point-to-point networks that are dedicated to a specific vendor's products, and also those that do not allow switched connectivity among attached devices. This eliminates networks such as IBM 3270 clusters, many of which satisfy the geographical constraints of our definition. Similarly, networks that simply reduce wiring costs through the use of statistical or other types of multiplexers will not be considered as LANs. A LAN should operate independently of any attached computer system, and should eliminate the need to establish separate transmission paths between all pairs of devices that need to communicate. The use of dedicated terminal-to-computer or computer-to-computer links for data communications is a viable alternative to either a PBX or a LAN, but these are a separate alternative. This is a very important alternative, however, since more data is transmitted over hardwired links than by either PBXs or LANs.

■ INFORMATION CHARACTERISTICS

Before evaluating PBX and LAN capabilities, the types of information that establish the requirements of local networks need to be examined. Local communications can be divided into 5 categories: data, text, voice, images, and

Data Transmission Alternatives: LAN or PBX?

graphics. In this categorization, the terms data and text are used to distinguish between data processing and office automation applications. That is, the term data is sometimes used only in reference to centralized data processing, while text is used in reference to applications such as word processing and certain types of facsimile. However, when one looks at network requirements, there is little distinction between these 2 categories. Both require similar bandwidths, protocols, and interfaces, and both types of applications are frequently supported on the same system. Thus, since the network requirements for transmitting data and text are almost identical in the many situations, we will use the term "data" more generically, to encompass both types of digitally encoded information. Any exceptions to this usage will be explicitly noted.

Since the vast majority of local communications traffic is either voice or data, we will look first at the needs of these 2 types of information. One dramatic trend in both local and long-haul networks is the transmission of digital voice. Digitization of voice in metropolitan networks grew rapidly during the 1960s, and began to penetrate intra-building networks in the 1970s (technology for local communications frequently originates in long-haul systems). This trend is stimulated partially by the advance of digital technology, but is also driven by the desire to transmit voice and data traffic over the same network facilities. Voice/data integration requires that voice first be converted from analog to digital format.

While it is feasible to send digital voice and data over the same transmission facilities, there are several other characteristics that are relevant in the analysis of local communication networks. For example, the bandwidth requirements of voice and data can differ dramatically. Data communications applications can be served by bandwidths of less than a hundred bits per second (bps) or can demand capacities in excess of a million bits per second (M bps). High speeds are needed to rapidly transmit large files, or to provide very fast response times to high-resolution CRT terminal displays. In contrast, most digital voice is transmitted at exactly 64,000 bits per second (K bps), although the standard adopted by Satellite Business Systems provides for voice at 32K bps. Thus, while voice channels all look alike, the information capacity demanded in data applications can vary by a factor of over 10,000.

Another important distinction between voice and data traffic is the consistency of demand on network facilities. During a typical digital voice connection, about 60% of the channel capacity is not used. Half of the available capacity is wasted because the talkers do not talk simultaneously, and the remaining 10% is lost because of the pauses that are a part of normal speech. The channel utilization for data applications is much more varied than for voice. For bulk file transfers on a half-duplex circuit, 100% of all the channel's capacity is used. In contrast, channel activity is typically only a few percent in interactive applications. Thus, both bandwidth and channel utilization suggest differing network requirements for voice and data traffic. This is illustrated graphically in Figure 1, which shows the range of bandwidth and channel utilization for voice, data, and video signals.

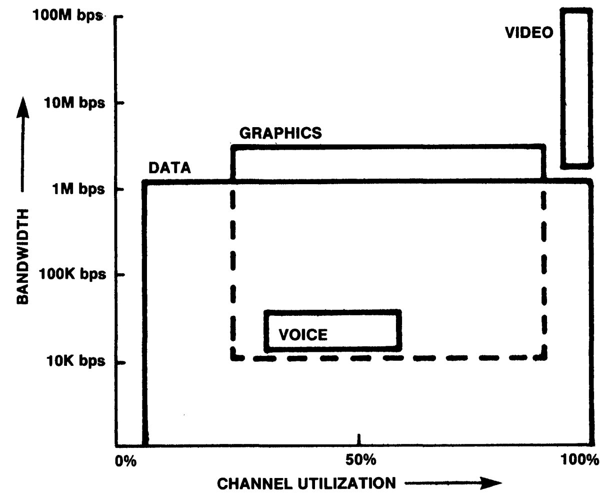


Figure 1 • bandwidth and channel utilization for different types of information.

Sensitivity to errors is an area where data requirements are much more stringent than those of voice. For example, digital error rates of one error every 10,000 bits would not be excessive for most voice applications but would be unacceptable for most data applications. Thus, data networks must be designed to have lower error rates.

The connect time for each session is another characteristic that distinguishes voice and data traffic. While the typical telephone call is only about 5 minutes long, the average data connection lasts much longer, and sessions that cover an entire working day are not unusual. All of these characteristics have important implications for the design of a local communications facility.

Another potential application of a local network is the transmission of images, which include all types of video: full motion, teleconferencing quality, and freeze-frame. Full motion video has the highest bandwidth requirement, needing approximately 100M bps for studio quality. This bandwidth requirement can be reduced to 45M bps using signal compression techniques, which reduces transmission costs at the expense of higher terminal costs. For teleconferencing quality video, the capacity requirement is typically reduced to the range of 1.5M bps to 3M bps. This reduction in bandwidth is achieved at the expense of signal quality and higher costs for video compressors. Finally, freeze-frame systems usually operate at much lower speeds, with the transmission time for an image increasing as the bit rate is reduced.

Video signal characteristics are also shown in Figure 1, where bandwidth and channel utilization are compared with that of voice and data. Video is somewhat unique in that the channel is almost always active 100% of the time. But, note that the video sector in this figure does not include freeze-frame, which is not widespread and would more appropriately fit at the top of the data sector.

The previous discussion of video was limited to digital signals to simplify the comparison with voice and data.

Data Transmission Alternatives: LAN or PBX?

However, the vast majority of video transmission is still analog because digitizers are still too expensive for most applications. Also, the very large bandwidth makes the transmission of digital video very expensive. Thus, the integration of digital video onto data networks must overcome 2 large hurdles. Data networks must have higher bandwidths, and digitizers must become less expensive. While progress in both of these areas will certainly be achieved, the advances required are so large that the integration of digital video onto baseband LANs will not be commercially important for the near term future.

The transmission of graphical information is the last category of network applications. Graphics are frequently intermixed with text and data, so the need to treat graphics as a separate category may not be obvious. However, the bandwidth requirements for this type of information can be much higher than for text or data, such that deserves separate attention. For example, the size of a typical terminal screen is 25x80 characters. The amount of data or text that can be displayed on such a screen is determined by multiplying these 2 figures, yielding 2000 characters (or bytes) of information. To obtain higher resolution, the information in a single screen must increase, allowing smaller elements of the display to be individually addressed. This higher resolution can be obtained through the use of "bit-mapped" graphics, in which the same screen could contain 26,000 bytes, or over 10 times as much. The use of color would increase the information requirement even further, and 100,000 bytes would be a typical number for business applications. In more advanced applications such as computer-aided design or imaging systems, the information requirements are even higher. As a final figure for comparison, a single screen of uncompressed color video transmitted digitally would require approximately 350,000 bytes of information. Although TV video is relatively low resolution, the coding scheme is not very efficient. Table 1 summarizes the above numbers, illustrating that the needs of graphical transmission are clearly greater than those of ordinary data or text.

Looking across all of the different types of information, distinctive differences in their transmission requirements are apparent. This illustrates why the integration of voice, data, video, and graphics onto a single network has proceeded so slowly. Thus far, no single networking technology has been economically attractive for all electronic communications. With this foundation, the following section will discuss some of the technological alternatives that are being considered as potential candidates for integrating these different types of information.

TYPE OF INFORMATION DISPLAYED	INFORMATION CONTENT (bytes)
Characters	2,000
Monochrome Graphics	25,000
Business Quality Color Graphics	100,000
Digital Color Graphics	300,000
Computer Aided Design Graphics	500,000 +

Table 1 • information content of different terminal displays.

■ LAN TECHNOLOGY

LANs can be categorized according to 4 different characteristics: media, topology, transmission technique, and access protocol. Each system can be specified according to these characteristics, although a few may encompass several subcategories. For example, the Ethernet specification includes provision for the use of coaxial cable, twisted pair, and optical fiber transmission media. Most systems will use only a single transmission technique and access protocol, but multiple topologies are not unusual. This section discusses the most common options that are available within these categories.

Media

Twisted pair, coaxial cable, and optical fibers are the 3 types of transmission media used in LANs, listed in order of historical discovery. That is, while twisted pair was in use before the turn of the century, optical fibers for communications have only been practical since the 1970s. Coaxial cable and optical fibers are generally perceived as being the most advanced candidates because they can transmit very high bandwidth signals. However, twisted pair has been used to transmit signals of 10M bps in both experimental LANs and in Ethernet drop cables. This is far less than is required in most applications. But, the distance between amplifiers is relatively low at high speeds, and the twisted pairs that are common in telephone systems could not be expected to support such bandwidths.

While twisted pair is frequently associated with voice communications, it is also used in many LAN products. Perhaps its most significant advantage is its pervasiveness, since twisted pair is likely to be already installed wherever there is a telephone. Another advantage is its relatively low cost, which varies from $0.04 to $0.20 per conductor foot, depending on the size of the wire and the number of conductors per cable. When compared with other types of cable, the ease of installing and splicing twisted pairs is an important attribute.

A key disadvantage of twisted pairs is their sensitivity to electromagnetic interference, or noise. This means that they can pick up spurious signals from other electrical equipment, such as computer systems, electronic motors, or even other twisted pair wires. Crosstalk, which is the electrical interference between twisted pairs in the same cable, is one of the important design constraints in high-speed digital transmission systems. Another potential disadvantage of twisted pairs is their signal attenuation at high frequencies, which can require the frequent use of signal amplifiers, or repeaters. However, since some LAN systems span short distances, this last concern is not always important.

Data Transmission Alternatives: LAN or PBX?

The transmission medium that is probably most frequently associated with LANs is coaxial cable. Three of the more widely promoted systems by Xerox, Wang, and Datapoint all use this medium. In contrast to twisted pair, coaxial cables have low susceptibility to electromagnetic interference and can transmit very high bandwidth signals. Technological maturity is another advantage of coaxial systems, since this medium is commonly used in electronic instrumentation, cable television, and long distance voice networks.

A significant disadvantage of coaxial cable is its bulkiness, which can lead to high installation and rearrangement expenses. It is also more expensive than twisted pair, in the range of $0.35 to $2.00 per foot, depending on physical and electrical characteristics. However, most of the cost of a LAN is usually in the electronic interfaces, common control equipment, and media installation, such that cable cost is not typically of overriding importance.

The type of cable that many observers predict will become more pervasive in future LAN systems is fiber optics. Fibers have higher digital bandwidths, lower signal attenuation, and greater immunity to electromagnetic interference than coaxial cable. In addition, they are smaller and lighter than twisted pairs. By these criteria fibers appear to have great potential, since they are faster, lighter, and more immune to noise. However, there are some important disadvantages.

A significant drawback of fiber technology is the precision and cleanliness required in splices and connectors. Fibers are very small (5-10 thousandths of an inch in diameter), and any misalignment or dirt at an interface can be catastrophic. Another consideration is the need for electro-optical transducers to convert from the electrical signals used in processing systems to the optical signals required by the fibers. Finally, the taps and couplers used with optical fibers' systems do not have the capabilities of those in electrical systems. While this makes fiber systems more secure, it also limits their use in certain network architectures. That is, fibers are not well suited for any topology that requires passive couplers. A fundamental constraint in multidrop optical systems is that the power levels in fibers are very low. This means that there is usually not enough power to share among several optical receivers.

Topology

Topology, or architecture, refers to the physical layout of a network. There are 4 main classifications of LAN topologies: bus, tree, star, and ring. These are illustrated in Figure 2. Note that this figure illustrates the physical relationship between the cable and the devices attached to the network, and is independent of how the wire is routed through a building. Intrabuilding cable tends to be routed along similar physical paths, independent of how the connections are made. Some observers of the LAN scene lump bus and tree into a single category, but there are good reasons to distinguish between these two. CATV systems use a tree topology, which is distinctly different and is incompatible with the bus architecture used in several LAN systems. In addition to these basic topologies, hybrids are possible. For example, a LAN development within IBM combines characteristics of both a star and a ring.

Looking at the more prominent characteristics of the different topologies, all signals are available at any access point on the network of a bus, tree, or ring network. This is convenient when all nodes on the network are to receive the same message (known as broadcast), but creates problems with network security. In contrast, a star is more secure, but the need for a separate cable from each device to a centralized hub requires more cable.

Considering network reliability, star and tree systems are more vulnerable, because a failure at the hub is catastrophic to network operation. Bus systems, in contrast, are typically designed such that a failure at any of the access points will not disrupt the network. Ring systems can be the most fragile, because each access point may be an electrical repeater, and the failure of any of these repeaters could bring down the system.

Transmission Techniques

The choice of a transmission technique is the LAN issue that has thus far generated the most controversy. Many debates, in both public and private forums, have tried to resolve the question of whether baseband or broadband is superior. Like many of the other aspects of LAN design, there is no single best answer. The optimal topology, media, access protocol, and transmission technique all depend on several factors, including geographic coverage, application, capacity, and growth requirements. Table 2 summarizes some of the key distinctions between baseband and broadband LANs.

The terms baseband and broadband are used primarily in reference to coaxial cable LANs, because this is the only media that is commonly used with both techniques. Optical fiber and twisted pair systems all tend to be baseband, because of constraints in the transmission capabilities of these systems.

The term baseband originates from the position of the signal in the frequency spectrum, implying it is at the low end, or base, of the spectrum. In reference to LANs, the term baseband implies that a single digital signal exists on the network at any given instant. This contrasts with broadband, where frequency multiplexing allows multiple signals to exist simultaneously. But, while baseband systems are purely digital, broadband systems use high-frequency analog modems, which are functionally similar to those used to transmit digital signals over the analog phone network. Among presently developed systems, broadband transmission typically provides greater information capacity and can cover a larger geographical area. In addition, some broadband systems can coexist on the same cable as a CATV system. If compatibility with analog video is important, then broadband is the only option. But, broadband interfaces can be more expensive and the networks can require greater maintenance.

Data Transmission Alternatives: LAN or PBX?

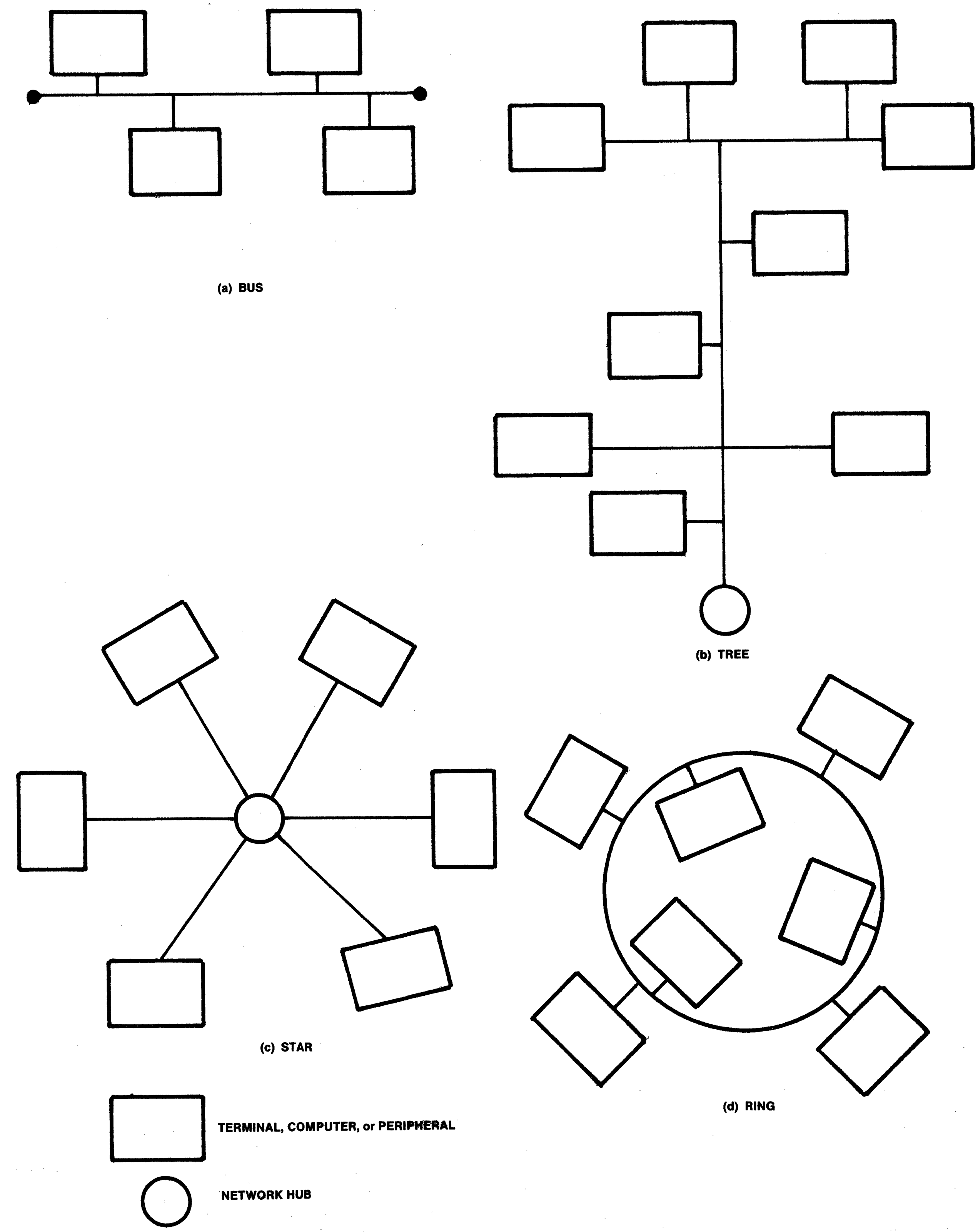


Figure 2 • LAN topologies.

Data Transmission Alternatives: LAN or PBX?

LAN Characteristic	Baseband	Broadband
Carrier Signal Format	Digital	Analog
Common Topologies	Bus, Ring & Star	Tree
Media	Twisted Pair, Coaxial Cable & Optical Fibers	Coaxial Cable
Number of Simultaneous Channels	One	Many
Analog Video	No	Yes
Distance Spanned	1000 feet - 3.75 miles	1.25 - 30 miles

Table 2 • characteristics of baseband and broadband LANs.

Access Protocols

In most LAN designs, several connected devices share the same transmission path. That is, each user puts his information into a frame, or "packet," before transmitting it over the network. When the transmission is complete, a separate user can send a packet. Under these conditions, there are different schemes used to determine how the network capacity is allocated among the attached devices. Access protocol refers to the rules that govern the order in which different competing devices can transmit their messages on the network. Token passing and carrier sense multiple access with collision detection (CSMA/CD) are the most widely used LAN access protocols.

In token passing, a special electronic signal gets sent throughout the network. When a device receives this token, it is allowed to transmit its message, and then sends the token to the next device. Token passing is known as a "deterministic" protocol, because the order in which devices gain access to the network is known in advance. Each device simply waits for its turn.

CSMA/CD is referred to as a "contention" protocol, because devices compete for network capacity. Whenever the network is inactive, any device can transmit. But, the transmitting device must continue to listen to the network, to determine whether its message collided with that of another device. If there was a collision, both devices must wait for a randomly chosen interval before attempting retransmission.

Considering protocol tradeoffs, the delay experienced by a device attempting to access the network is more predictable in token passing systems than in CSMA/CD. While the delay is greater in token passing under light loads, it does not degrade excessively as the network approaches its capacity. In contrast, the access delay is probabilistic in CSMA/CD and could get unacceptably large when the load offered to the network approaches its capacity. Also, with CSMA/CD the probability of collision increases in larger networks because it takes longer for devices to become aware of network access attempts from more distant devices. Given that most LANs today are relatively small, these debates do not yet have profound practical significance. But, since networks are expected to grow, it is important to recognize the limitations of different techniques.

Looking at the spectrum of LAN characteristics, there are several choices facing a potential user. With 3 possible media, 4 topologies, 2 transmission techniques, and 2 access protocols, 48 combinations are possible. This occurs before considering subcategories and hybrids. But media, topology, transmission technique, and access protocol cannot be chosen independent of each other. Vendors have settled on a few common designs, and these are listed in Table 3. As each product category has different capabilities and constraints, it is essential that users understand the requirements of their applications before committing to a specific design.

■ PBX TECHNOLOGY

PBXs were originally designed to serve only the needs of intrabuilding voice communications, using technology similar to that used in the public switched network. Each channel needed 3K Hz of analog bandwidth, a public network standard that was originally selected because of constraints in early telephone microphones. Today, however, the needs of data transmission, coupled with advances in semiconductor technology, have stimulated dramatic changes in PBX design.

While there have been profound advances in PBX design over the last decade, the associated intrabuilding wiring architecture has remained fundamentally the same for over 50 years. Twisted wire pairs are used to connect each individual telephone to a centralized switching location.

Vendors	Topology	Media	Transmission Technique	Access Protocol
3-COM, Xerox, AST, Ungermann-Bass	Bus	Coaxial Cable	Baseband	CSMA/CD
Datapoint, Lanier	Bus	Coaxial Cable	Baseband	Token Passing
Intersil, Wang, Sytek	Tree	Coaxial Cable	Broadband	CSMA/CD
Proteon, Racal-Milgo	Ring	Coaxial Cable	Broadband	Token Passing

Table 3 • common LAN configurations.

Data Transmission Alternatives: LAN or PBX?

Early switches were constructed of large arrays of electromechanical relays, which provided physical connections among the attached telephones and to the public network. Since the mid-1970s, semiconductors have rapidly replaced electromechanical devices, and computerized common control has provided for a large number of new features. Of the 4 network topologies discussed earlier, essentially all PBX systems are configured as a star. As potential exceptions, Ztel and CXC have announced products that will switch voice over distinctly different architectures, but it is too early to judge either the technical or market success of these ventures.

The need to transmit data over analog voice facilities led to the development of modems (modulator/demodulators). The same modems that transmit data at speeds from 300-4800 bps over the public voice network can be used to transmit data through a PBX. But there are some important limitations to this approach, such as error performance and available digital bandwidth. To overcome these limitations, PBX manufacturers have turned to digital switching techniques.

In digital PBX networks, voice channels get converted into digital signals. In the most widely used format (PCM for pulse code modulation), each voice signal becomes a 64K-bps bit stream. Once a network is designed to handle 64K-bps digital signals, data can be transmitted as easily as voice. Thus, data transmission speeds can be over 10 times greater with digital PBXs than with conventional analog systems.

While the above discussion implies that a PBX is either analog or digital, there is also a hybrid technique that is common. Pulse amplitude modulation (PAM) is a cross between digital and analog, in which voice is converted to discrete analog pulses. Data capabilities of PAM systems are very close to those of analog PBXs, and the importance of PAM is rapidly diminishing. Digital offerings now dominate the large end of the PBX spectrum (greater than 1000 lines), and the availability of smaller systems is also increasing (large systems were the first to become economic using digital technology).

Within a PBX network, voice is typically digitized at one of 2 different locations. In early systems, the A/D (analog/digital) conversion typically occurred at the central switch. With this technique, the signals that were generated by a telephone and transmitted over twisted wire pairs were still analog. More recently, digital telephones have become available, potentially eliminating the need for any analog transmission in the network. In this latter case, not only can voice and data be handled by the same central switch, but integrated transmission can exist over the same pair of wires. Different vendors have their own versions of integrated voice/data protocols, with the bandwidth per wire pair varying from 160 to 256K bps. This trend toward voice/data integration is expected to accelerate in the future.

Most digital PBX vendors have developed high speed interfaces that allow several voice or data signals to be multiplexed over a single channel. The standard that has evolved from the public voice network is known as T1, which provides for 24 simultaneous 64K-bps channels over a 1.5M-bps facility. With T1 interfaces being developed for PBXs, these networks will have an efficient high-speed interface to either the public network or to other private networks.

■ NETWORK COMPARISONS

A comparison of the data transmission capabilities of LANs and PBXs must consider several independent issues. Among the important parameters are protocol compatibility, network capacity, distance, response time, wiring, growth flexibility, voice/data integration, and systems management. The choice of a specific network will depend on the needs of each user, since no single technology is optimal for all situations. Using as a foundation the previous sections on network applications and technologies, this section will compare the suitability of LANs and PBXs for meeting different user needs. With continued changes expected in the local communication arena, both present and future needs will be considered.

Protocol Compatibility

One of the most important issues in choosing a network is its compatibility with the devices among which the users wish to share information. To attach a device to a network, both must share a common interface, which refers to the electrical and mechanical characteristics of the point of interconnection. In addition, the network and attached devices must share common protocols, which are the binary coded symbols used to set up and maintain order in the flow of information. Unfortunately, there are many different incompatible protocols, and the lack of compatibility among different computer vendors is a major barrier to the sharing of information in distributed processing environments. It is not even unusual to find major incompatibilities within the product line of a single vendor. Thus, in selecting a network, it is important to consider the protocols and interfaces of the devices that will be attached. The work of standards organizations is expected to simplify the interconnection problem in the future, but for now, compatibility between devices and networks is a major user concern.

The ability to allow the interconnection of previously incompatible devices is frequently mentioned as a major goal for LANs. That is, in addition to allowing connection between devices that could previously communicate, many view LANs as providing protocol conversion between types of computer equipment. But, while this goal is certainly highly desirable, it is also very difficult to achieve. Thus far, any protocol conversion associated with LANs has occurred outside of the network. The value of the network in this case is to allow several terminals access to a single protocol converter. Similar configurations are available with PBXs, in which the PBX provides shared access to converters that allow terminals to be used with noncompatible computers. Common conversions are from ASCII asynchronous to IBM's Bisync, and ASCII async to SDLC.

Without protocol conversion, networks only allow communication between devices that were already compatible.

Data Transmission Alternatives: LAN or PBX?

Different classes of devices may still share the same network, but they can only communicate among other devices of the same class. Still, not all classes of equipment are compatible with all networks, so the following section will discuss some of the issues regarding the compatibility of LANs and PBXs with some of the more common computer protocols and interfaces.

The most common class of communication interfaces for computers and terminals is RS-232C, which encompasses several distinct implementations with varying levels of functionality. There is no general statement to make regarding the superiority of PBXs or LANs in allowing attachment through this interface, because there are greater variations within these product classifications than there are between them. All networks can be expected to support the basic RS-232C functions required by most terminals. But since there are variations in the support of the less frequently used functions, users should be aware of the differences. An interesting implication of the preponderance of RS-232C is the bandwidth limitation it implies. That is, with its 19.2K-bps limit, this interface can be the bottleneck in gaining access to the full bandwidth potential of a network.

While compatibility between interfaces is the first step in allowing attachment to a network, there must also be compatibility between the protocols transmitted through these interfaces. There are many different types of protocols, and before discussing this issue, it is important to consider how the different protocols are related. The International Standards Organization (ISO) has been working toward the development of standard protocols that would serve as a basis for connecting different vendors' equipment. Given the difficulty of this problem, a preliminary step has been the development of a reference model that establishes a framework for grouping the different types of protocols. The result of this work is the reference model for Open Systems Interconnection (OSI), which divides data communications protocols into 7 distinct categories, or layers. This layered architecture was formally approved in May 1983, has since been widely endorsed, and the development of specific protocols within the model is now underway.

In comparing LANs and PBXs, it is the first 2 layers of the ISO reference model that are most important. Minimum data communications service demands that network and devices be compatible over the first 2 levels. The first level encompasses the interface specifications and is referred to as the physical layer. The second level is known as the data link layer and includes the protocols for transferring data over a single link. The RS-232C interface discussed earlier falls in the first layer, so we will now discuss some general issues related to network connectivity in the data link layer.

The protocol adopted as a standard for layer 2 in the ISO reference model is HDLC (Higher-level Data Link Control). Different vendors have their own version of layer 2, and some examples include SDLC (Synchronous Data Link Control) by IBM, BDLC by Burroughs, UDLC by Sperry, and DDCMP by Digital Equipment Corporation. IBM's Bisync is another protocol family that includes many layer 2 functions. Looking at the ability of PBXs and LANs to interface with different layer 2 protocols, it is important to recognize that the previously mentioned HDLC family of protocols are all synchronous. Unfortunately, it is easier for packet data networks to support asynchronous transmission. This applies both to public packet-switched networks such as Tymnet and Telenet, and to many local area networks. Broadband LANs can easily support synchronous protocols because they can offer a dedicated channel, but baseband networks are not as flexible. One reason for this problem is that baseband networks must share the same channel among many devices, and any single device cannot be guaranteed access to the network at any given instant. As another problem, the HDLC type protocols are full-duplex (simultaneous two-way transmissions), but baseband networks are not inherently full-duplex. Both of these problems can be somewhat overcome by using very high speed baseband transmission, allowing users to be given the impression that a synchronous full-duplex channel is available. But the implementation of such interfaces is both more difficult and more expensive than that needed for asynchronous transmission.

Although baseband LANs have some constraints in the transmission of synchronous full-duplex protocols, their adaptability depends on other design factors. For example, networks using token passing can be more easily adapted for synchronous transmission than those using the CSMA/CD access to protocol. The reason is that the delay in gaining access to the network is much less predictable for CSMA/CD than it is for token passing. This problem can become particularly acute as the load offered to the network approaches the maximum capacity of the network. One of the motivations for IBM's choice of token passing in their LAN is the need to effectively handle SDLC, which is a synchronous full-duplex protocol.

For PBXs, synchronous transmission is not a major problem. During each session, a user has a dedicated full-duplex path through the system. Once a connection is established, the network is essentially transparent to the format of the information passing through the switch. While early digital PBXs only offered asynchronous communications, all major vendors have since announced support of synchronous transmission.

The upper layers of the OSI reference model cover the remaining protocol issues that must be addressed for the full sharing of information between the computer applications across multiple network modes. The details of these other layers are not important here, except to note that software is now available from several LAN vendors to provide the functions needed in these upper layers. This software typically resides outside of the LAN on another computer, usually a micro, and is designed to be fully compatible with the lower level LAN functions. With this software, users have all the tools necessary to share information among several processing elements attached to the network. Such packages offer far greater functionality than that available with PBXs, which do not interact with the information they transfer.

Data Transmission Alternatives: LAN or PBX?

A user problem that has created significant interest in LANs is the rewiring of IBM 3270 environments, which have a user base in excess of a million and a half terminals. In the most popular configuration each terminal has a dedicated coaxial cable to a cluster controller that is co-located with the mainframe computer. It is not unusual to find over a thousand such terminals in a single building, and some companies have estimated the rewiring costs to move one 3270 terminal to be as high as $2,000. With terminals moved as frequently as every 1-2 years, large savings could be achieved with a pervasive network that would allow terminals to be unplugged and then plugged back in. Much development work has been directed toward the solution of the 3270 problem, but thus far neither PBX nor LAN vendors have developed interfaces that can integrate large numbers of IBM 3270 terminals onto their networks (other vendors have offered more flexible solutions for non-IBM terminals). A partial solution to the 3270 problem has been achieved through the use of remote cluster controllers. A remote cluster can be connected to the host computer through a relatively low-speed SDLC link. In this case, the previously discussed constraints regarding synchronous data transmission are applicable. Unfortunately, there are cost and performance penalties associated with this latter configuration, and most users have not found this solution to be an attractive alternative to locally attached cluster controllers.

Network Capacity

A frequently mentioned feature of LANs is high bandwidth, since the information capacity provided to each attached device significantly exceeds that offered by PBXs in most cases. However, it is important to distinguish between total network capacity and the instantaneous capacity available at each access point. It is in this latter capability that LANs excel. While digital PBXs offer from 64 to over 200K bps to each attached workstation, the instantaneous bandwidth from LANs is more typically in the range of from one to tens of megabits per second. This high bandwidth is important when transferring large files quickly or for interactive applications that require both high screen resolution and fast response time.

Shifting from bandwidth per attached device to the total capacity offered simultaneously to all devices, digital PBXs have much greater capacity than LANs. The total capacity of a baseband LAN is typically the same as the maximum instantaneous bandwidth offered to each device, which ranges from one to tens of megabits per second. Broadband LANs can have significantly greater capacities, easily exceeding 100M bps, depending on frequency allocations and sophistication of interface equipment. Looking at large digital PBXs, total capacities of several hundred Mbps are more typical, although much of this capacity would likely be devoted to voice. As mentioned previously, with most LANs still relatively small, total capacity is not yet extremely important.

Although the total digital capacity of a PBX can be very large, there is little flexibility in the way this capacity is allocated. In contrast, LANs are optimized for the traffic characteristics of data applications, in which the attached equipment is much more diverse. For devices connected to LAN, bandwidth is allocated on demand, and devices with low channel utilization only deplete network capacity while data is actually being transmitted. This is distinctly different from digital PBXs. Whenever a terminal or computer has an active connection through a PBX, it depletes the network capacity whether or not data is actually being transmitted. In addition, the amount of network capacity that is depleted for most systems is the same as that required for a single digital voice channel. This implies that a duplex 300 bps data connection through a PBX would deplete network capacity by 128K bps, again independent of whether or not data is being transmitted. This is reasonable for voice, since all channels require approximately the same bandwidth and channel utilization. But, it can be very inefficient for data requirements. Thus, although the total digital capacity of LAN is usually less than that of a PBX, LAN is more efficient in how the capacity is allocated and provides higher instantaneous bandwidths. The fast response times required in many data applications demand this latter feature.

Distance Constraints

The geographical dimensions of LANs vary widely across different products. In general, broadband networks can span greater distances than baseband networks. For example, broadband networks have been designed to span distances of up to 50 miles, although 3-8 mile limits are more common. Thus, it is doubtful that distance limitations of broadband networks will be significant for intrabuilding or even campus environments. For baseband networks, maximum distances are more likely to be from about 1000 feet to a few miles, again varying for different products. Distance limits for Ethernet, a widely recognized product, are 1500 feet using only coaxial cable, and 2500 feet using a combination of optical fibers with coaxial cable. Note that distance constraints could become important for baseband networks in very large buildings or in multi-building complexes.

An important feature of baseband LANs using the CSMA/CD access protocol is that the distances spanned are reduced as the total network bandwidth is increased. Network efficiency is also closely related to both of these parameters. With CSMA/CD, the longer the network, the longer it takes for a device to become aware of other devices' attempts to gain access to the network. Thus, the probability of collision increases for longer networks, degrading network efficiency. Also, for higher speed systems, there are more bits simultaneously on the network, which means that more information is lost from collisions. For PBXs, distance constraints are rarely of major concern. Telephone stations can be up to a few miles from the centralized switch, and the end-to-end distance is twice this figure. Also, newer systems allow several PBXs to be networked together, such that even greater distances can be covered.

Wiring Issues

At present, the most common media for LANs is coaxial cable, and for PBXs it is twisted pair. Many commercially

Data Transmission Alternatives: LAN or PBX?

available LANs also use twisted pair, and preliminary information from both IBM and AT&T indicates that each of their products will use this type of medium in addition to incorporating optical fibers. Looking at the pervasiveness of different media across both PBXs and LANs, twisted pair is by far the most widely installed cable type, existing in essentially every building for telephone service. This widespread use of this media is a significant motive to the use of PBXs for data transmission, since it may frequently be the only alternative that does not require additional building wiring.

Considering wiring differences across LAN products, almost all twisted pair and optical fiber products use baseband transmission. Also, baseband networks tend to use ring, star, or bus topologies, while broadband networks typically use tree structures. Note that there is much more diversity in the use of different media with LANs than with PBXs. The use of optical fibers to allow attachment to PBX subsystems at remote locations has only begun to become available. As another interesting factor, while different PBX products will all work with the same twisted pair (presuming sufficient pairs to link to each terminal), no similar statement can be made about LANs.

Interest in building wiring is frequently most intense when new construction or renovation is being planned. A significant factor in wiring a building is that the installation cost may far exceed the cost of materials. Since this installation cost can be minimized during construction, many users would like to use such an opportunity to implement a wiring plan that would serve all future applications. Unfortunately, there is no easy answer to this question at present. The only certainty is that twisted pairs will be required for voice. If one knows the specific type of LAN that will ultimately be installed, one can choose the medium that is compatible. But, continuing changes in LAN technology make futile any attempts to choose a LAN that will satisfy all present and future applications. And, since LANs use a variety of media in several different topologies, there is no single media that will be compatible with the diverse array of LAN products. Two examples to illustrate the current situation are the incompatibility of baseband coaxial cable with broadband products, and of broadband topologies with baseband systems.

Integrated Transmission

Before discussing the integration of different types of information onto a single network, it should be noted that there are 2 distinctly dissimilar perceptions of LAN capabilities. On the one hand, LANs are occasionally portrayed as the ultimate solution for local communications, integrating all types of information onto a single facility. This differs dramatically from the reality of most present installations, which typically have only from 10 to 20 devices attached, and are limited to the transmission of data, text, and graphics.

There are 2 fundamental reasons for combining different forms of information on the same network. One important motive is cost reduction by reducing total cable requirements, eliminating expensive wiring rearrangements, and avoiding duplicate overhead for network maintenance and administration. A second motive is improved functionality; that is, new features can become available in integrated systems. Given that the fraction of users that are committed to totally integrated networks is relatively low, it can be concluded that neither of these advantages has yet been overwhelmingly demonstrated.

When examining the possibility of integrating voice, data, graphics, and video, baseband LANs are the least flexible. While voice can be digitized relatively easily, the network capacity of most baseband systems is not sufficient to support extensive voice traffic. For example, Xerox has experimented with a voice version of Ethernet, referred to as Etherphone. But this system uses different protocols than the original Ethernet, such that it requires a separate cable. Even with a separate cable, the system will support only 45 simultaneous conversations. As a final constraint, the delay introduced by the system under heavy load can degrade performance when connections are made to the long-distance network. While this is only one example, it is generally true that baseband LANs will not be very useful for integrating real-time voice and data onto the same network. Voice for store-and-forward systems is more likely to be developed first, because response time is not as important.

Digital PBXs are much more adaptable for voice/data integration than baseband LANs. Early systems typically transmitted voice and data over separate twisted pairs, but newer systems actually multiplex both signals over the same copper conductors. However, while integrated voice and data is common in PBX systems, voice networks are not necessarily efficient in serving the long holding times and low channel utilizations that are characteristic of data traffic. Thus, data intensive environments could excessively deplete the capacity of PBX networks.

Although PBXs can integrate voice and data, video transmission over these networks is not likely for the near-term future. PBXs are not designed to handle the large bandwidth requirements of video. The only possible candidates for the integration of voice, data, and video are broadband LANs. Since broadband networks typically use the same type of cable as CATV systems, it is not at all difficult to combine video and data communications on the same network. Such systems are becoming increasingly common. But including voice communications on these same networks may not occur for several years, if ever. Although the bandwidth and the technology exists for transmitting voice on broadband coaxial systems, the economics are not yet attractive in local networks. While experimental systems have been available since the late 1970s, no commercial developments have yet proven successful.

Network Administration & Control

Network administration and control is a phrase used to describe several features that are related to how a network's resources are managed. This includes issues such as obtaining traffic statistics, recovering from system failures, security, establishing user priorities, and managing growth

Data Transmission Alternatives: LAN or PBX?

and rearrangements. These are the type of features that are important to network administrators, for it is their job to insure that a network remains operational and that a system can grow to serve the capacity demanded.

It is generally true that the software available for collecting traffic statistics and for diagnosing system failures is much more advanced for PBXs than LANs. Similar monitoring functions could be implemented in LANs, but most of these products have not been around long enough to have had such features developed. But with an installed base in the billions of dollars, PBX vendors have been able to make greater investments in the advanced software needed for administration and control functions. Also, since most LAN installations are very small, the need for advanced administrative features is not very great.

The ability to assign user priorities is a feature that varies widely across LAN products. Baseband CSMA/CD systems are not typically designed to deal with different user priorities. All users are treated equally, and access to the network is purely statistical (i.e., priority goes to those who are lucky). Token passing schemes are generally more flexible in varying user priority. Advanced PBXs are also able to assign different priorities, giving preferred users both first access and more features.

Security of information is not an area where LANs excel. In general, all signals on LANs are available at any access point on the network. Thus, it would be very easy for a user to tap in and listen to all network activity. Data encryption would be needed to insure secure information. Security on PBXs is greater because each terminal has a dedicated path to the central switch, and the wire into each user's office contains only the signals intended for that user.

Flexibility for growth is a major concern for users choosing a network, as the need for data transmission continues to increase at a rapid pace. Thus, the ease of adding terminals to a network, expanding network coverage, and increasing network capacity are all important issues. The ease of adding a new node can depend largely on whether the network has already been cabled to the desired access point. Given that cable already exists, baseband bus systems can be tapped into without disturbing other users on the network or without the intervention of a network administrator. Token ring structures can be more difficult to add a station to, since the network may have to be taken out of service to add a terminal. With PBXs, additions are relatively simple, although the network administrator must get involved to make changes at the central switch. Considering other growth constraints, the previous comments about system capacity and distance limits continue to apply. That is, both PBXs and broadband LANs can grow to much larger networks than can baseband bus systems.

Network Costs

It is difficult to make any general statements about LAN costs because there are such great variations from one product to another. Quoted costs per connection can range from a few hundred to tens of thousands of dollars, although from \$500 to \$2000 is most common. Also, cable costs may or may not be included, and cable installation is typically left to the user. As another relevant issue, low connection costs are occasionally achieved by sharing a single network interface among many attached terminals, which is not useful unless the terminals are in the same area of the building. Consistent with all types of equipment, users can expect to pay more for higher bandwidth, greater functionality and better service. Compared with voice systems, digital PBX connection costs are more predictable, and start in the range of about \$1000 per port. Both PBXs and LANs are likely to benefit from further cost reduction, but as with any new technology, it is essential that users pay close attention to what a product can and cannot accomplish for a given price.

Other Alternatives

The discussion thus far has focused on the trade-offs of PBXs and LANs for local data communications. Two other alternatives also need to be mentioned, the data PBX and hardwired connections. Looking first at the data PBX, this is a network that architecturally looks very similar to a voice PBX except that it is used only for data. These systems allow a large number of terminals to share computer access ports, and also allow terminals to gain access to several different computers. Like voice systems, no higher-level network software is provided with these products. Advantages of the data PBX include a relatively low cost per port and the use of inexpensive twisted pair. The main disadvantage of these systems is that they are designed around the RS-232C interface, which implies only moderate bandwidth per port (19.2K bps) and limited distances. Thus, while they may effectively solve many user problems today, their growth potential is limited.

The final and most common option for local data transmission is to run a dedicated cable between the 2 pieces of equipment that wish to share information. The number of such connections far exceeds the number that is switched over a PBX or LAN. The advantage of this approach is that it avoids the cost of an independent network and is inexpensive if most terminals only need to be connected with a single computer. A major disadvantage is inflexibility, since there is limited sharing of transmission resources. Since any networking software must be resident in the attached devices, this method encourages independence between different vendors' equipment. Thus, as information must be shared among more systems and as the number of connections grows, the fraction of data transmission that occurs over hardwired connections can be expected to decline.

■ IMPLICATIONS

After examining the major issues related to the use of LANs and PBXs for local communications, some significant conclusions can be drawn. First of all, there is much more diversity in data transmission equipment than in voice equipment. One reason for this situation is that most telephone technology has been developed inside monopolies, public or private, where standards could be enforced. In contrast, the data processing industry evolved in a very

Data Transmission Alternatives: LAN or PBX?

competitive environment, with divergent designs and capabilities reflecting different user applications. Also, the functions and capabilities of data processing equipment vary much more widely than those of the human voice. Looking at the available networking products confirms this situation, with a wide range of designs used in LANs, but with little variation in the fundamental design of PBXs. To the extent that the diversity in LAN products reflects differing user needs for data communications, the development of standards can be expected to proceed slowly.

Although there has been significant progress toward LAN standards on the IEEE 802 committee, if one looks beneath the surface, there is much that remains to be achieved. Standards have been established for 2 network categories, the baseband CSMA/CD bus and the token passing bus. Token rings and broadband architectures have yet to be resolved, which include both Wang's product and IBM's anticipated product. Also, AT&T has hinted at LAN development that is distinct from all others. Even within the IEEE 802 standards, the medium remains unspecified, which allows for a number of variations within each category. Thus, the situation regarding LAN standards is that it is likely to be a while before the field gets narrowed significantly.

In comparing the PBX situation with that of LANs, while there is greater architectural similarity among voice networks, the digital protocols used to connect terminals to the central switch are different for each manufacturer. Also, voice communications remain the primary motive for a PBX purchase. For data transmission, users are likely to consider a PBX as only one possible alternative on a long list of alternatives. As with LAN products, there are certain situations in which a given alternative may not have the desired functionality or may not be cost-effective. An added handicap facing PBX suppliers is that many users do not wish to integrate their voice and data onto the same network, for either technical or organizational considerations. The environment where PBXs are most attractive is where data is to be transferred among a few devices scattered throughout a building, where it is not likely to be cost effective to make major wire additions for a small number of attachments. For concentrations of equipment in a specific area, the cost-attractiveness of LANs increases.

Given that no single LAN is likely to serve all data networking applications, the likelihood of a single network integrating all voice, data, graphics, and video communications within a single building is even more remote. Thus, while LANs are occasionally viewed as the answer to a single universal wiring plan, this ideal is a long way from realization. Similarly, while some expect LANs to solve the problem of incompatibility between different systems, protocol conversion typically takes place outside the network. Both LANs and PBXs provide physical connection with minimal protocol conversion.

In the multivendor environment facing most users, compatibility remains an overriding concern. The existence of LANs does not solve the compatibility problem, but merely changes its character. For the forseeable future, there will be several incompatible data networks. This will force the development of interfaces to these networks and of gateways between networks.

PBXs will not displace LANs, nor will LANs displace PBXs. For the immediate future their markets are more distinct than they are overlapping. From this perspective, the problem facing both LAN and PBX vendors is to insure that their products can effectively interconnect. Since it is doubtful that interfaces can be built between all possible combinations, intelligent selection of candidates will be essential. Similarly, for data processing and office equipment vendors, interfaces to other equipment and other networks will be important. In an era of integrated information management systems, no single vendor can do it all, and even IBM and AT&T have shown a willingness to pursue joint ventures with other manufacturers. In this environment, an important element of success will be the cooperation of carefully selected partners.

• END

PC-to-Mainframe Communication

■ INTRODUCTION

The increase in use of personal computers in corporations with existing data processing facilities has already raised questions on the coexistence of distributed processing power in PC form and central data centers. The existence of local PC data collection naturally opens questions on the ability to capture PC data directly for data center applications, or even to use PCs in conjunction with traditional DP equipment for efficient data collection. But there are major questions of data processing policy to be resolved in PC use. One of the most critical issues is the maintenance of data control in such a mixed environment. Many data processing experts fear that PCs may weaken central management information systems by diverting data from the corporate database, or that dissemination of information to so many separate processing points will result in local distortions or inconsistencies.

Providing a link between personal computers and the corporate data center can be a significant step in optimizing PC usage and in controlling the integrity of the total corporate information base. To perform effectively, however, the communication link must bridge the very different environments of personal computer and minicomputer or mainframe. This report discusses the problems in connecting personal computers to mainframes, alternative connection technologies, and factors which affect the selection of an ideal link between PC and data center.

■ PERSONAL COMPUTERS AND DATA DISPERSION

Today's business environment is increasingly dependent on the productivity and strategies of its professional, technical, and managerial personnel. While automation in support of these individuals is not a recent development, the advent of personal computer systems has made it possible to provide key people with a locally controlled level of computational power which equals that of an entire data center of twenty years ago. The explosive growth of personal computer usage by key corporate personnel can be attributed to corporate recognition that such power, used in support of planning and operational decisions, can reduce costs, increase profits, and free decision-makers to apply themselves to other projects in support of corporate goals.

The microcomputer has also altered the price/performance relationships on which the decisions for clerical automation have been based. Data entry productivity can be increased manifold by providing local screen formatting and data entry; and local systems for such functions as document retrieval can often reduce department clerical effort significantly. Personal computers based on current microprocessor technologies are often less expensive than terminals into data center mainframes, and package software which supports the users' needs can be purchased from a retail store rather than developed at considerable time and expense by corporate data processing organizations.

There is a price for the productivity gains of personal computers, however, and it is often paid in the area of Information Resource Management. A corporation's operating information base is itself a significant asset, and management information systems rely on proper maintenance of the basic operating data to provide summaries of current activities and to project future trends. Personal computers can "shortstop" vital information before it reaches the data center, making it unavailable to corporate MIS. A department may be able to maintain its own budgets more efficiently on a local personal computer, but is that gain in efficiency worth the loss of the information to the business as a whole?

Even local departments may suffer from some problems associated with local personal computer data storage. User organizations normally lack the DP department's familiarity with the needs for proper editing of information and controls to insure reliability and consistency of data. It may be perfectly obvious to a local department supervisor that an expense report date cannot be in the future, but unless that fact is related to the program gathering expense information, that restriction cannot be assured in data being entered on the file. Reports which display expense data in current-month, last-month form may fail to show the item at all, and it will almost certainly be in the wrong place when it eventually does appear. Lack of a central collection point for data may result in multiple "versions" of information in the personal computer files of several departments, or even in several systems in the same department. These inconsistencies and editing problems can be disastrous if attempts are made to collect and use the information at a later date.

■ USING DATA COMMUNICATION TO HALT THE "DATA STAMPEDE"

A logical answer to many of the data integrity and consistency problems, and to the need for central information management and control, would appear to be connection of the personal computers in use within a corporation to that company's data center through some form of communication link. The connection could then be used to gain access to primary data bases in the data center, eliminating the need for local storage of information. A file could be loaded to the personal computer when needed and returned when updated, or a direct dialog between data center file manager and personal computer user could be accommodated for item-by-item access and update.

Personal computers and the data center can relate to one-another in several basic ways, as shown in Figure 1. First, they can link together using a "batch" protocol when required to exchange files. This process substitutes communication paths for a form of compatible storage media such as disk or magnetic tape. If a file could be carried from a PC to the data center on a floppy disk, file

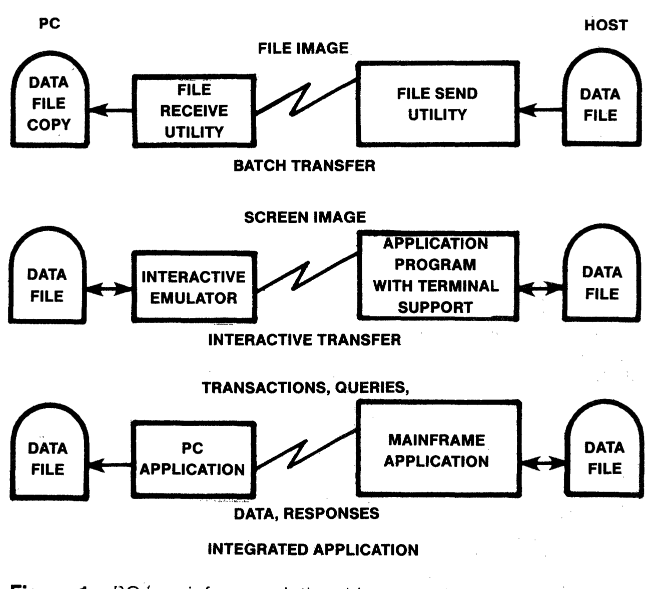


Figure 1 • PC/ mainframe relationships.

PC-to-Mainframe Communication

exchange would actually be easier than this form of communication. A second option is to attach the PC by emulating an interactive terminal on the data center system. This gives the local PC operator a means of interacting as a terminal user, and may be extended to provide programs on the PC with the ability to interface through the "pseudo-terminal" as well. File transfer may still be accomplished through terminal emulation if the data center software runs a form of terminal-to-disk copy program. The final option for attachment of a personal computer involves software on the host which works with software running on the PC to provide a pathway between systems. Data and even requests for processing can pass over the pathway, and the PC/data center form a single unit co-operatively working to satisfy the user's needs.

Ideally, corporations would like microcomputers and data center equipment to relate to each other in a way which would make the exact location of the data and the processing power being harnessed for any particular user task to be transparent to the user. If this is to be made possible, there are some basic rules about the data link between the systems which must be enforced:

- The communication technique used must be compatible with the support currently available at the data center, or at least must be supportable with upgrades to that equipment.
- The personal computer must emulate a form of data interaction which is acceptable to the communication software in use or available. A data center which supports only interactive terminals, for example, may have difficulties supporting personal computers operating in batch mode even if the hardware at the data center would permit the connection.
- The information exchanged must be in a form accessible for processing at the destination with a minimum of additional processing. This may mean code set translation since most personal computers use the ASCII character set and IBM mainframes use EBCDIC.
- The operation of the communication link must be as transparent to the non-technical PC user as possible, and must disrupt the normal flow of the PC application as little as possible to preserve the local ease of use which probably justified the use of PCs in the first place.

The best approach to providing a facility such as a PC/data center link is to examine the functional requirements and select a method which satisfies them within reasonable economic restraints. This approach may be excessively idealistic for the micro/mainframe communication problem because the data center equipment tends to be a "given" because it is already in place, and the capabilities of microcomputers in communication may be very limited. Given these restrictions, it is probably best for the user to evaluate the characteristics of the data center environment and the microcomputer communication environment to establish the framework within which a flexible communication facility can be developed.

■ THE MAINFRAME COMMUNICATION ENVIRONMENT

Data center equipment, being more expensive and supporting the mainstream of corporate data processing, normally establishes the basic communication environment. While some accommodation to personal computer communication can often be made, the cost of major changes to the data center communication facilities can often exceed the cost of personal computers. If PCs are to successfully communicate with the data center, the first step is to determine the communication parameters which the PCs must satisfy to do so.

Data communication support in existing facilities can be classified according to the type of exchange of information supported. Most modern data centers which provide any form of communication support do so for INTERACTIVE TERMINALS such as CRTs. These devices provide the means for direct user entry of data into the central systems, a substitute for keypunch and other means of off-line data collection. Some data centers may also support BATCH transmission of information, either because off-line data collection is still in use or in order to support direct file transfers from computer to computer. Often batch and interactive communication use different communication protocols, but even where the protocols used are compatible, batch and interactive communication may involve different communication software in the host system. Application software will almost certainly be different. A major bank which uses CRTs for data entry can interface very easily to a personal computer which can appear to the data center as a CRT because the software used for the application is designed to deliver data summaries in response to a terminal inquiry. The same system, presented with a batch of inquiries in a single transmission, lacks the software to produce a batch of status reports as a single-transmission response. MOST SUCCESSFUL PC-TO-DATA-CENTER CONNECTIONS WILL EMULATE APPLICATION COMMUNICATION DIALOGS WHICH ARE ALREADY SUPPORTED BY THE DATA CENTER. In all cases, it is important to know the communication software and application level support available at the data center before considering any particular method of PC communication.

Given the goal of emulating some form of existing application dialog, the setup of a PC to data center link must consider the following characteristics of such a dialog:

- **Protocol.** Most minicomputers and mainframes have a preferred communication protocol—IBM for example normally uses its System Network Architecture (SNA). The PCs communicating to mainframes MUST use a protocol which is supported by the mainframe, but SHOULD use the particular protocol already in use at the data center. Some systems, particularly the larger mainframes, are somewhat terminal independent within the group of terminal types supported. These systems may even permit the substitution of terminals of one protocol for those of another within the APPLICATION-LEVEL dialog. For these systems, protocol can be a secondary factor in selection of a communication technique, as long as the protocols being considered can all be supported in the same way at the application level.
- **Attachment mode.** Conventional point-to-point RS-232 cables or phone circuits and modems are most often used to attach PCs to data center computers. Some types of equipment may be usable in multipoint mode on circuits already supporting native terminal devices. Other more exotic methods of connection are sometimes available. IBM systems may support the attachment of PCs to 3270 controller devices directly via coaxial cable, and DEC computers may be able to accept Ethernet attachment of some personal computers. Most of these alternative attachment methods offer significant advantages, either in data rates or economy, and should be seriously considered where available.
- **Facility layout.** Personal computers will rarely be used in close proximity to the data center, so consideration must be given for the impact of the relative locations of the equipment. This is particularly important where attachment other than via conventional phone/modem means is contemplated. RS-232 cable has a nominal limit of 50 feet and even the extended distance cable is rarely usable beyond 250 feet. While other interface methods such as RS-449 may offer greater distances, these may not be supported either at the PC or on the data center equipment.
- **Speed.** The data rate of the data center communication equipment may be set by the requirements of other terminal devices or connected computer systems. If the data rate is high (above 1200 pbs) it may restrict some personal computer attachments because popular PC modems and other interface devices normally operate at 1200 bits per second or less.
- **CPU port capacity.** Most data centers have a few spare hardware ports for the attachment of new communication devices, but there may be several types of ports, each with their own restrictions and characteristics. The cost of adding new ports to a CPU or data communication controller may be quite high, especially in the case of protocols such as SNA or IBM's older binary synchronous protocol. You may want to review the CAPACITY of the data center system for communication ports of each type, as well as for the number of currently available spares. It would probably not be wise

PC-to-Mainframe Communication

to dedicate the last SNA port in an operation heavily dependent on SNA communications to a personal computer!

While a data center environment may be restrictive in terms of its ability to support PC communication, it is at least normally designed for SOME form of communication support. Data center computers are usually designed to support multiple users and to share system resources among a large number of separate "tasks". Adding personal computer connections to such an environment is largely a matter of matching communication capabilities on both ends and of allocating the required resources. If the economics of the PC link application justify it, the data center can shift policy somewhat to accommodate a reasonable PC interface method. The key question is often whether the PC can supply such a method.

■ MICROCOMPUTERS AND DATA CENTER COMMUNICATION

There are three primary factors to consider in personal computer architecture when data connections to a host computer are required:

- The ability of the personal computer to support communication functions at the same time as it supports local use.
- The support for communication protocols and terminal functions which match the expectations of the data center environment.
- The level of integration between PC application software and data center software to support co-operative tasks and to permit transparent file sharing.

Data communication is a function which is often called "real-time," meaning that communication activities are almost like human activities; they tend to occur in response to outside influences which cannot easily be scheduled or predicted, and once they occur they must be handled promptly. A microcomputer system which serves a local user may or may not have the ability to concurrently serve a communication link to a host. Microcomputers of the type normally used as personal computers or executive workstations were designed primarily as standalone systems. The operating systems used on the systems probably do not support multiple users or multiple "tasks," actions which are logically separate and could take place concurrently but are associated with a single user. The difference between single-task and multitask systems can be seen in Figure 2. A single-task computer can undertake only one function at a time—word processing, spreadsheet, database, or communication. The user must decide when to invoke each of these functions, based on current needs. This type of serial scheduling of work is normally suitable for human interactions; people are also typically single-thread in work habits. But even with non-communication environments there are times when you probably could have used a quick access to a spreadsheet program in the middle of writing a large report on a word processor. To get it, you must probably exit the word processor, saving your work, and enter the spreadsheet program. Communication interactions are quite different. Communicating between micro and data center is not an end to itself, but a step in the handling of information which has company-wide meaning. As such, it may occur as a PART of any other job and not likely as a job in itself. Putting it another way, communicating between personal computer and data center is naturally an INTERACTIVE kind of task. Single-task personal computers, on the other hand, are BATCH oriented. Requests for communication service are normally made outside the application, leaving the user with the problem of integrating the data received with local work or of extracting and formatting data to send.

Multiple 1=1 task systems, as shown in Figure 2, address this problem by making the communication program coexist with the application. Services can be requested at any time directly from the user's application program. Since this permits a program to request communication directly, it supports the transparent use of communication links to the data center as a part of any user application, should that application be written to use such a link. Multiple-task communication may be provided on personal computers which are designed for multiple-task operation, or vendor software may establish a separate "communication" task in an

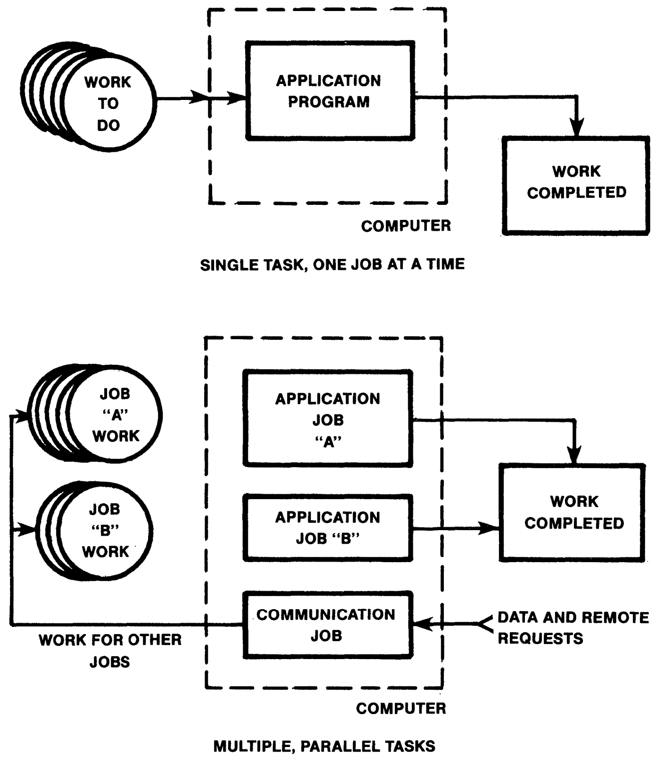


Figure 2 • work flow in single and multitask systems.

otherwise single-task environment. Figure 3 shows the structure of a popular 3270 terminal emulator system available for the IBM Personal Computer. The system permits the user to "install" a copy of the emulator software in the personal computer memory ALONG WITH THE USER PROGRAM BEING RUN. This resident copy can then be used by programming languages such as BASIC to access data center information, or the operator can key a special sequence of characters to switch between the emulator and the user program.

Both multitask personal computer systems which permit a continuously active communication function and "co-resident" communication programs, may permit the operator to switch keying and display between local programs and emulation of a data center terminal. This permits the operator to interlace local and remote functions, but only by acting as the common element in two dialogs—one with the local application on the PC and the other with the data center application. If PROGRAM-LEVEL interchange between data center and PC is desired, not only must the communication support on the PC provide facilities for it, the application programs must use those facilities. It is this factor which makes multitask operating systems or other forms of full-time communication support from the PC vendor superior to any form of "add-on" communication support. Software developers will not undertake an incremental expense to support the new XYZ Corporation communication board and software unless they believe that a significant number of personal computer users to whom they target their product will buy and use XYZ Corp's board. The PC vendor's standard features have a good chance of gaining widespread support, and optional features which prove popular have some chance. It is very unlikely that a significant number of PC application programs will support communication capabilities offered by a third-party vendor.

TECHNICAL FACTORS OF THE COMMUNICATION ENVIRONMENT may have a significant impact on PC connections to the data center. This is particularly important for users of non-IBM, non-DEC systems, since most popular "terminal emulation" systems

PC-to-Mainframe Communication

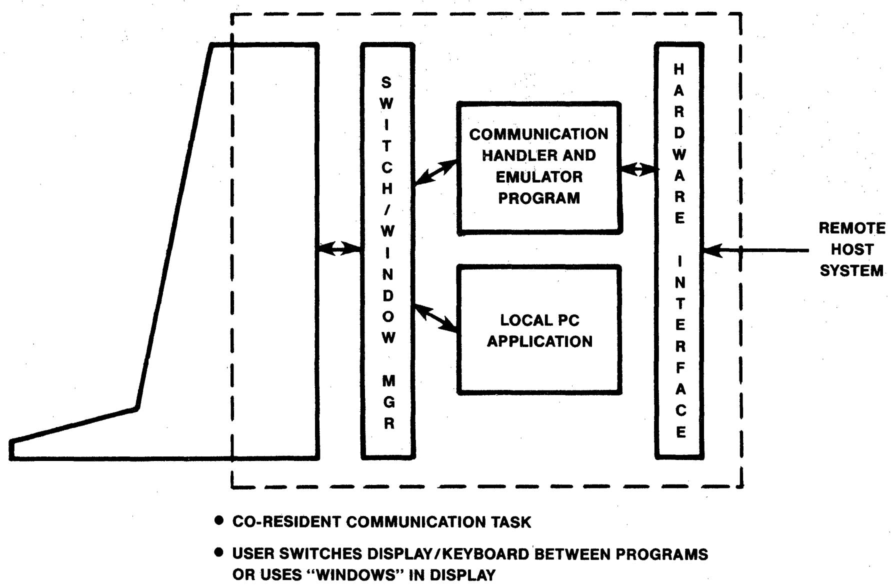


Figure 3 • a popular terminal emulator structure.

tend to emulate IBM or DEC terminals. The physical support for communication on a micro must suit the data center to which connection is to be made. That support is often envisioned as a "board"—hardware additions to the personal computer with the proper interface for the communication system involved. This is rarely true. ALL PRODUCTS WHICH SUPPORT CONNECTION OF PCs TO DATA CENTER COMPUTERS WILL REQUIRE A SOFTWARE COMPONENT. Selection of a package for communication support must be made by evaluating both the hardware and the software elements of the package. A package which consists only of hardware is logically incomplete, since SOMEONE must provide program support for the actual communication. If the vendor does not, you are elected. Packages which consist of only software are more common, and may be useful if the hardware on which they depend is either standard equipment on the personal computer being used or a commonly-used option.

Hardware options for PC attachment to the data center are shown in Figure 4. First, the communication hardware provides the means of physically attaching the link to the host computer and controlling the signals on that link. This attachment may take several forms:

- A standard communication interface such as RS-232C, used to attach the PC to a host system directly via cable or through a modem.
- A coaxial cable or other special interface to a COMMUNICATION CONTROLLER, which allows the PC to attach as part of a cluster of terminals. The IBM 3270 display system attachment of CRTs to a cluster controller is an example of this type of connection.
- An interface to a vendor-proprietary or industry-standard bus or local network architecture. Examples of this type of interface are the various Ethernet connections or the IEEE-488/Hewlett-Packard Instrumentation Bus.

Selection of a personal computer attachment hardware architecture is normally dictated by the capabilities of the data center. Where several options exist, the flexibility of the approach (communication interface via modem is suitable for many environments while bus connection is a local option only) versus the performance and convenience of the interface (coaxial or bus attachment may support much higher data rates and require no additional CPU communication ports or software support) must be evaluated.

OPERATIONAL ISSUES IN PC LINKS TO MAINFRAMES may outweigh the technical ones. A personal computer acting as a mainframe terminal is normally designed to emulate the functions of the terminal as well as the technical characteristics. This may mean that the PC user will have a set of operating rules for the stand-alone environment, established by the local software which is normally used, and one or more additional sets of rules associated with each mainframe and perhaps each application interaction on that mainframe. A New York bank PC operator, having spent the morning switching between 3270 emulation to an IBM host and a local spreadsheet package was heard to remark, upon sitting down at the computer after a break, "Let's see—who am I now?"

■ MICRO-MAINFRAME LINKS—WHAT ARE MY CHOICES?

A user faced with the need to select a means of linking personal computers to the data center is confronted by choices in several areas:

- **Operating strategy.** The PC can emulate a terminal to an existing application on the mainframe, with or without support for direct transfer of files. It can also link with the mainframe in a joint application supported by custom software in both places.
- **Mode of attachment.** The PC can connect directly via data cables, usually an RS-232 interface, through a modem, via coaxial cable to a cluster controller, over a local-area network, directly to the computer bus, or in other mutually supportable ways.
- **Product source.** The equipment and software needed can be acquired from a single source or multiple sources, and those sources can be computer vendors, modem vendors, third-party hardware or software suppliers, or system integrators.
- **Information management and control.** Figure 5 shows some of the forms of PC data interaction which can take place, and the impact on data center files. PC data can be maintained entirely on data center files and updated only through data center applications. This restricts the PC to a data manipulation and reporting role. That role can be expanded by allowing local data entry and processing to take place, with reconciliation of information with the common data bank taking place either interactively or

PC-to-Mainframe Communication

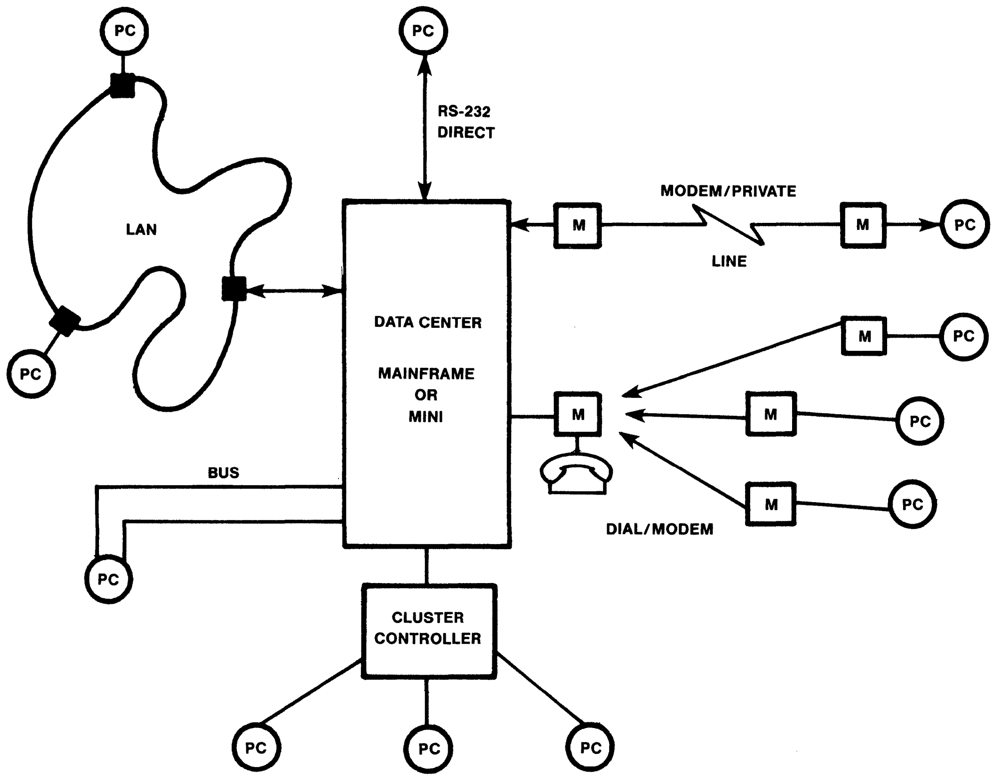


Figure 4 • hardware connection options.

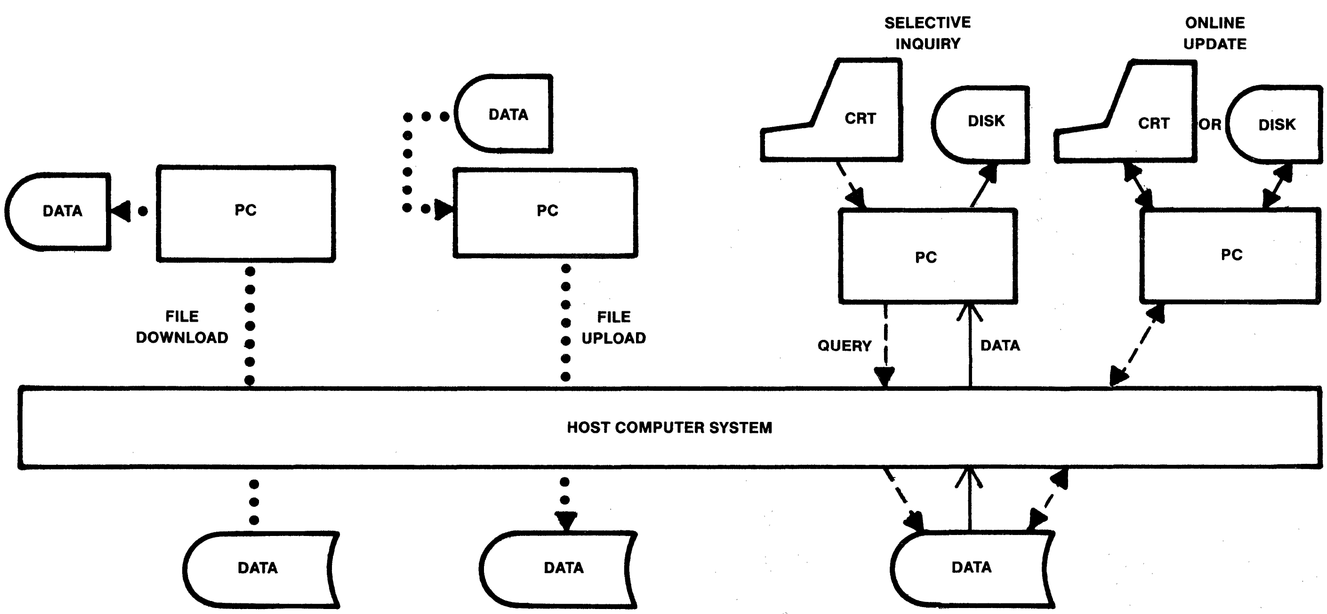


Figure 5 • file interactions with remote PCs.

when the data is ultimately loaded into the data center files. If local update of existing data center files through an upload-modify-download process is also to be permitted, some form of protection must be applied to prevent concurrent updates of the same data and to warn other users of information that a copy is outstanding for update purposes.

PC-to-Mainframe Communication

■ TERMINAL EMULATION SYSTEM

Most users will probably find that connection between their microcomputers and the data center will be supported by having the microcomputer emulate a terminal which is supported by the data center. The key questions in such applications are the type of terminal to emulate and the way in which local applications can utilize the emulator interface.

Most users have some form of CRT terminal on the data center equipment already, supporting online applications or programming. Emulating one of the existing terminal types would permit attachment with a minimum number of changes to the data center communication software, a task which often makes PC considerations pale into insignificance. IBM computer systems will normally be best served by the emulation of interactive display terminals of the 3270 family or the 5250 family. DEC systems operate best with terminals emulating the VT52 or VT100 terminals. Some IBM users may also find products which emulate the batch workstations such as the 2780 or 3780 suit their operating environment, but such devices are less common than the interactive terminals.

Once the type of terminal has been defined, or a range of acceptable devices has been identified, the issue of connection mode can be applied. If the PC and the mainframe are relatively close together, a simple interface cable can be used to connect them. BE CAREFUL ON DIRECT CONNECTION WHERE THE EMULATED TERMINAL HAS A SYNCHRONOUS PROTOCOL. SYNCHRONOUS LINKS REQUIRE A CLOCK SIGNAL, AND NOT ALL PC TERMINAL EMULATORS CAN SUPPLY A CLOCK. You may need a modem eliminator on such connection. IBM 3270, 2780/3780, and 5250 devices are ALL synchronous.

Modem connections will normally work between the PC and the mainframe, but the type of modem will depend on the distance. Limited-distance or short-haul modems, sometimes called line drivers, will serve best for connections within a building. Your existing telephone wiring may be able to handle everything, or new wires may be needed. The type of modem and its speed must match the terminal being emulated. Try to supply the highest speed supported by both PC and mainframe—nobody ever complains that response time is too fast!

IBM 3270 emulators may directly connect to the host computer using either the binary synchronous or SDLC/SNA protocols, or they may attach via coaxial cable to one of the 3270 family's cluster controllers. The coaxial attachment method may reduce your installation costs considerably if you have a cluster controller in the area of the PC. Check with the PC and cluster controller specifications to be sure that the cable run is not too long, that the terminal type emulated by the PC package is supported, and that the host software has been modified to add the new terminal address if required. You may want to consider buying a new cluster controller just to attach PCs if there are a large number expected from a single area of the building. No modem is required to attach a PC to a cluster controller via coaxial cable.

DEC users may find that Ethernet attachment to the DEC host is an attractive option, providing that DECnet support for Ethernet is available in the data center. In cases where Ethernet is already used or where a large number of PCs will be emulating DEC terminals, Ethernet may be an economical connection option. Several vendors supply personal computer links to Ethernet, and the data rates supported by these products are high. Since Ethernet attachment is not as common as the more traditional RS-232 or modem attachment, the ability to select products based on features will be restricted.

Terminal emulator products will rarely emulate EVERY aspect of the target terminal. Keyboard layout is seldom flexible on PCs, so emulators are rarely able to do more than approximate the actual device. This may result only in small adjustment factors for operators, but sometimes terminals have a bank of function keys which cannot all be duplicated on the PC. Loss of some of these keys may affect the operation of some applications. Screen display characteristics, particularly in graphic mode, may also differ, making the emulator unsuitable for some specialized functions. If the target terminal has a status line, be sure that it is emulated in an understandable form. It is important to review OPERATOR factors and APPLICATION factors as well as technical factors in selecting a terminal emulator product.

The most significant problem in the use of terminal emulators may be in the area of data control. When a PC is used to capture a segment of a mainframe data base and move it to local storage, that data becomes subject to manipulation OUTSIDE THE APPLICATION FRAMEWORK WHICH WAS INTENDED TO VALIDATE IT. In many cases, the movement of data to a PC, its editing, and the return to the host will bypass any host editing or validation. Even if the data is sent to the PC by an application program and returned by that same program, there is a good chance that some of the changes made while the data was "remote" will not be fully edited on its return. The host database is thus gradually polluted with questionable information.

The problem is magnified if several PCs are allowed access to the information. A major insurance company had an application where file segments were sent to personal computers via a batch terminal emulator and processed locally. There was no protection against several PC users calling for the SAME FILE SEGMENT, and when the data was returned to the file only the last user's changes were reflected.

There are some steps which can be taken to reduce the risk of corrupting data center files through the use of personal computers emulating terminals.

- Limit "out-of-data-center" updates. If personal computer copies of files are used only for access, there is no danger of poor edit control producing bad data.
- Require that update applications be reviewed by the data processing organization to certify that the data quality assurance measures taken by the data center system are also applied at the PC.
- Provide a "staging" area where PC files which have been updated are returned for review and validation, PRIOR TO BEING ACCEPTED INTO THE MAIN DATABASE.
- Initiate a form of access control on data which can be loaded into PCs to prevent multiple users from requesting the same data elements at the same time.

SELECTION OF A TERMINAL EMULATOR should consider the following points:

- Does the product emulate a terminal type and protocol which are already supported at the data center, or which can be supported easily?
- Will the method of attaching the product to the data center equipment be satisfactory given the location of the systems and the types of wiring, etc, which are available?
- Is the product compatible with the PC or PCs used and with any special hardware or software which is already in use on the systems?
- Is the mode of operation of the product suitable for the level of operator on the PC and for the type of work which is expected to be done with the product?
- Are the operating procedures so different from those associated with the PC's local operation that they will be difficult to complete properly?
- Can the combined environment created by the product between the PC and the host be controlled from the viewpoint of access security, data integrity, and information resource management?

IF SEVERAL PRODUCTS ARE GENERALLY SUITABLE FOR USE, the following guidelines may help narrow the choice:

- If the PC vendor supplies a package which meets your needs, give it preference over comparable products from third parties. The chances of the vendor's own product tracking future changes in the PC line are better than those of a third-party product.
- If the communication hardware supplied by the PC vendor is supported by a software-only product supplied by a third party, give that combination preference over another

PC-to-Mainframe Communication

product where both hardware and software are third-party products.

• Avoid products which rely on the combination of third-party software from one vendor and hardware from another third-party vendor. How many support people do you really want to talk to?

• Select a product with an interface to an external modem over one with similar features but using an integral modem. Modem technology is changing very rapidly, and the non-modem product will track those changes if you desire.

• If a product contains an interface board which plugs into the PC directly, buy from a company which supplies many types of such boards over one who supplies only the board used in the product. Broad familiarity with the PC's bus architecture will translate into fewer potential problems in coexisting with other board-level products.

• Select products which ENHANCE the capabilities of the device emulated without requiring host support for those enhancements. Integral printing support for terminals which cannot normally drive a satellite printer is an example of this. Beware of buying on gimmicks, however.

■ THE INTEGRATED APPLICATION SOLUTION

Terminal emulators have several significant disadvantages. Their basic design causes them to emulate a device which possesses no local processing or storage power, and therefore the interactions which the host supports with them will not easily take advantage of such capabilities. Terminals also have logical design restrictions, limitations in functionality which can be safely accepted since the human operator's range of supported functions is also limited. The limitations may be unacceptable when a PC is substituted for the terminal. For example, a PC emulating a 3270 terminal device probably cannot transfer binary data files because binary data is not presentable to an operator and is not supported as input to or output from most interactive terminals. Extension of the power of the PC through such devices will normally require special programs and operational controls in the data center to permit free data exchange and preserve data integrity.

The problems with projecting PC power can be solved by applying some of the principles of distributed processing to the PC environment. If software in a personal computer can link to similar software on a mainframe, the resulting INTEGRATED package can work to satisfy user needs somewhat independent of the exact location of the resources which must be applied. Figure 6 shows an example of the structure of an integrated software program package designed to permit personal computer users to interact with their data center mainframe. In contrast to the terminal emulators, which place the human operator of the PC into communication with the host application, the integrated applications provide an "application manager" element at both ends of the connection. This element-pair manages the communication path on behalf of the two systems and provides the services of data exchange and process requests to either system. The complexity of such an environment is directly proportional to the complexity of the applications managed, so cost constraints act to limit the scope of integrated software systems. MOST INTEGRATED SOFTWARE PACKAGES ARE LIMITED TO ONE OR A FEW APPLICATIONS, AND THE MORE SOPHISTICATED A SINGLE APPLICATION SUPPORT FACILITY IS, THE FEWER SUCH FACILITIES ARE LIKELY TO BE PRESENT.

The operation of the integrated application can be understood by reference to Figure 6:

• The operator makes a request at the PC keyboard, which is received by the local application manager. The manager determines that the request is one for a local PROCESS and directs it to the proper software element in the PC.

• The PROCESS ELEMENT determines that it requires data for the request and that need is communicated back to the application manager, who determines that the data is not local but resident in the mainframe.

• The application manager requests its opposite number running in the mainframe to provide the data. The mainframe application manager makes its own local PROCESS ELEMENT request to do so, and sends the data back over the communication link.

• The PC application manager delivers the data to the PC PROCESS ELEMENT, and that data is used to satisfy the user request.

Remote service requests can likewise be transferred over the communication path to the host system, and the data which was supplied by the host can be stored locally for use later in the same session. If the data is updated rather than just accessed, the PC

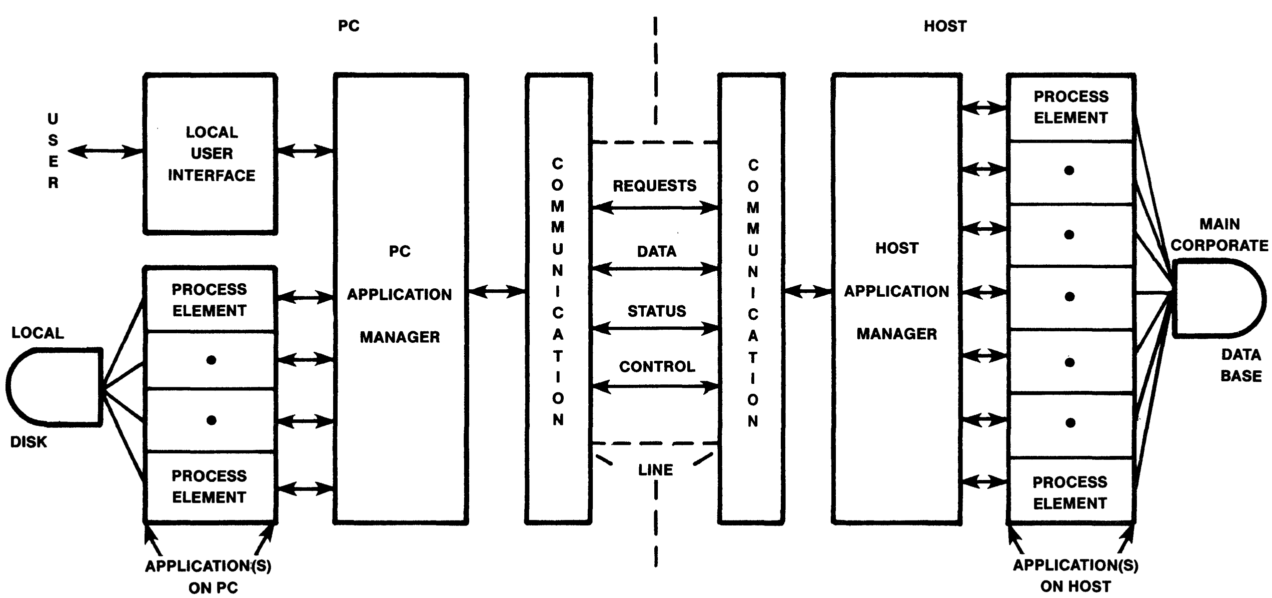


Figure 6 • a tightly-coupled integrated application.

PC-to-Mainframe Communication

application manager can route the update through to the mainframe, where all the proper controls for changing the database can be applied.

In some systems the structure of the software and the flow of information is much as described above. Other systems use a "loose" structure, where an application manager at each end of the connection acts as a "data filter" between the PC and the mainframe. Figure 7 shows this type of system. In it, a PC user makes a data request, often as a "query" in a database language. This query results in a request for data, which is satisfied by downloading information from the host to a PC file. That file is then processed locally on the PC using standard programs and uploaded when update of the mainframe data base is desired. This style of integration is more easily implemented, but often offers the user little more capabilities than a terminal emulator. Since data is NOT exchanged interactively, the problems of data integrity and loss of operator attention due to changes in job context are likely to remain.

Most users find this concept of application integration so attractive that they wonder why anyone would ever elect to use another method of PC communication with data center systems. The answer is that integrated applications are extremely rare, a fact that relates to the differences between such applications and the normal PC and mainframe software.

Figures 3, 6, and 7 which show the comparative structure of a terminal emulator application and an integrated application illustrates these differences. While the PROCESS ELEMENTS of the applications in the host and in the PC may be generally similar between the two architectures, those of the INTEGRATED APPLICATION must be designed to operate on requests which are generated internally by the software and not through the action of a terminal operator. Each PROCESS ELEMENT in an integrated application must be very modular in structure, because it may perform only a small part of a total task which may span several computer systems. It is more difficult to design and program such structures, and therefore more expensive. Personal computer software has developed in a simple, single-user environment, and most of the popular software available is not easily adapted to integration with mainframe software, a fault which is equally applicable to the mainframe products. Some progressive vendors are already producing integrated systems, and others are working to reconcile the two computer environments with modified or totally new software. The market trend seems to be clearly taking the direction of the integrated application.

TECHNICAL COMMUNICATION SUPPORT SELECTION with integrated application software is usually a matter of following the requirements stated by the supplier. Most of these packages will NOT support wide varieties of communication protocols, modems, or custom interface boards, so selection of hardware in advance of selecting the integrated package could create a problem. There is also a danger that two separate integrated applications may require different hardware support. Users would be advised to prioritize their needs for such software and watch for incompatibilities in supporting hardware. In general, the systems and hardware with the best chance of having a wide range of compatible integrated applications are those where the PC vendor itself has provided proper technical facilities for mainframe communication. Software vendors, given a consistant and preferred hardware environment, will tend to build to it. The IBM PC is an example of this; IBM's own host communication support is the preferred attachment method for most of the PC's integrated application software.

DATA VALIDATION AND CONTROL in an integrated environment may be totally within the control of the package, making data handling at the PC as safe or safer than the same process at a mainframe terminal. The potential for this level of validation exists with integrated packages, but the promise may not be fulfilled in practice. As the level of integration decreases, from the "ideal" tight coupling shown in Figure 6 to the looser structure shown in Figure 7, the ability of the application to manage the data flow and to insure that proper validation rules are applied at each update decreases. One area of concern is the ability to perform local data manipulations on data resident in the PC, either as a normal practice for loosely integrated applications or in tightly integrated applications when the communication link is down. This may be a useful feature from the standpoint of immediate production requirements, but its availability opens the door to divergence of the local and central databases. Tightly integrated applications which provide almost transparent host access and full distribution of files and computer resources are likely to be restricted to a single application, but offer the highest level of control over the information resources of the company. Loose structures may be applied to more PC and mainframe software so the processing flexibility provided is high, but the potential for corruption of data through poor control on the PC is significantly higher. Users must evaluate the benefits of wide application flexibility against the potential loss of control.

EVALUATING AN INTEGRATED APPLICATION can be

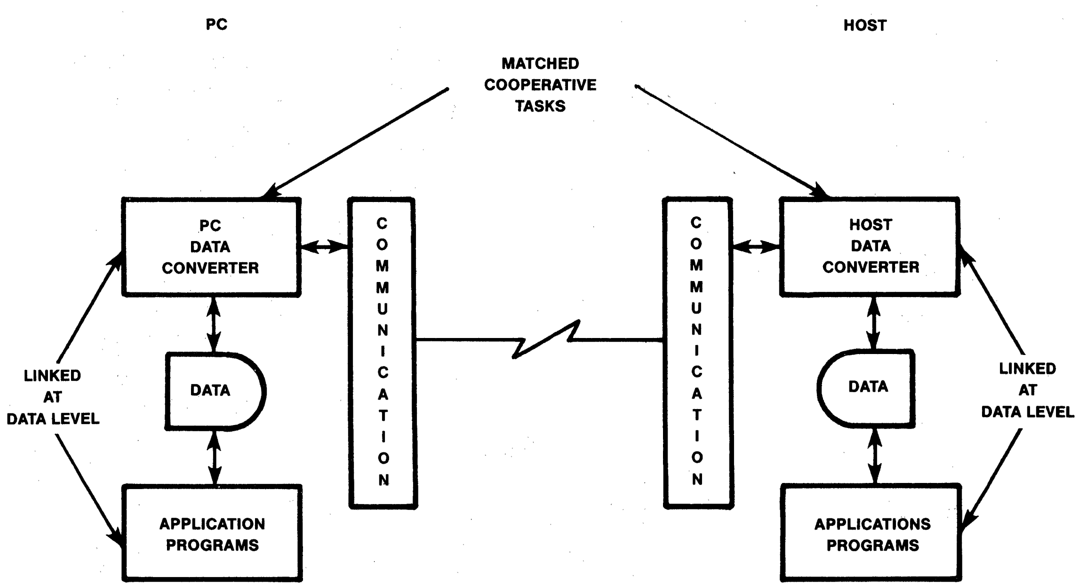


Figure 7 • a loosely coupled integrated application.

PC-to-Mainframe Communication

approached in phases. During the first phase, the user identifies "target applications" for such systems, and indicates the host and PC software (if any) which are already in place in the application areas. The second phase requires the construction of an "ideal access" scenario, where the system's operation is described as the user would like to see it work. These phases provide the user with an idea of the current environment for the task and the investment in it, and of the way in which the task should operate, economic constraints aside.

The third phase of evaluation is the actual investigation of integrated application alternatives. These alternatives may be identified based on competitive survey publications, advertising, or contact with software vendors. When a list has been compiled, each product should be subject to an examination based on the following considerations:

- **Host hardware and software requirements.** The corporate investment in any mainframe packages already in place in the application area is likely to be impressive, and the value of archival data accumulated through the life of the present system may be very high. Integrated software which requires a major change in the host environment, while sometimes justified, is usually just too expensive an option to consider.

- **PC requirements.** The penetration of PCs into the corporate environment has proceeded faster in some organizations than others. Companies and divisions with a large PC investment will have to consider the impact of each potential integrated application on their PC users. These impacts may range from fundamental issues of compatibility (it won't work with the existing computer base, or it will) through configuration questions such as the addition of memory or disk storage to each system, to operational questions relating to the software already used or the way in which the systems are run. Each package considered should be rated in its impact on the PC environment in the key areas of basic compatibility, hardware configuration requirements, adaptability to currently used software, and operational considerations.

- **Level of integration.** Packages will vary according to the degree to which the PC and host software is tied together. In a tightly integrated package the PC user will make requests without knowing where the data or resources needed to carry them out actually reside. This type of application will limit the ability of the PC user to operate with non-integrated software, and may in fact prevent use of the integrated files with any other software package. Loosely integrated applications will generally provide only a facility to download data from the host, translate it into a usable PC format for local processing with many different software packages, then permit its uploading to the host to replace or update the database. Highly integrated packages are easier for the users to handle because they present a consistent, request-oriented operating environment, but they are almost always functionally limited. Choosing a level of integration involves measuring the degree to which the package corresponds with the ideal model of user interaction developed in the previous phase. This must then be balanced against any loss of functionality or compatibility with existing software which is associated with the package.

- **Data security and information resource control.** This is perhaps the most important aspect of the evaluation process, and the most difficult to apply. DP management should normally be asked to provide input on the level of security provided with each application.

■ MAKING A FULL EVALUATION OF MAINFRAME COMMUNICATION ALTERNATIVES

The first step in any selection of mainframe communication package is to GET THE DATA CENTER MANAGEMENT INVOLVED. They should provide the information on the current data center configuration, hardware, software support, application programs, and data security measures.

Once the proper specialists have been assembled, the application should be evaluated in BOTH terminal emulation and integrated application terms. A good way to do this is to "walk through" the interactions associated with the application to get an idea of what the PC user will experience. You will probably need the input of the data center professionals to help with the way in which the terminal emulator will interact with the data center equipment. You may want to do three "walk-throughs"; one for interactive terminal emulation, one for batch terminal emulation, and one for integrated applications. Don't try to get too specific to any package here; just get an idea of whether the GENERAL interaction is acceptable.

The next step is to evaluate the data security and information resource management implications of the PC link. Ask the following questions:

1. Are requests for data made from the PC subject to satisfactory levels of access control, such as password control? The controls over PC access should be at least as good as those controlling the access of terminals which could normally be expected to query the data.

2. Can similar control be provided at the PC for operator access to data once the data is downloaded? It is undesirable to protect data in the data center and leave it available to all the world once it is loaded onto a PC.

3. Is it possible for several copies of the data to be outstanding in different PCs at the same time? Can this be controlled in any way at the host end? Multiple copies of the same file are invitations to divergence of information. If you cannot prevent a file from being accessed by several users at once, it may be better to require that the file REMAIN IN THE DATA CENTER and be accessed only for USE and not for STORAGE at the PC. Integrated applications may offer a solution to problems which come to light during discussions on this issue.

4. Can the return of data to the mainframe result in the loss of any update controls? These may arise from a lack of control in the PC over the update process or from the inability to provide all the transaction editing at the PC due to lack of other key data files. In any case, a relaxation of the validation rules may result in a lowering of data quality standards, and this should be SPECIFICALLY UNDERSTOOD AND APPROVED BY MANAGEMENT.

5. How long will information be "out of file"; resident in a newer version on the PC than in the main database? Long intervals when the data center does not have current data are HIGHLY UNDESIRABLE, so if data is to be held on the PC for more than a day or so the application should be reviewed. The data center should check the possible impacts of "out-of-file" data on key reports which are run periodically for corporate management or external distribution, and conflicts may indicate that PC production schedules will need adjustment for the period near the report cycle.

The walk-through and the security and control evaluation should provide a list of requirements for the mainframe communication link, which can be added to the technical features of the PC and the data center equipment for the purpose of completing an evaluation of existing packages. The most suitable can be examined in detail or even benchmarked for final selection.

PC/MAINFRAME INTEGRATION must eventually be addressed if personal computers are to be properly used in a corporate environment. The potentials offered by widespread data exchange between the small computers and the data center in terms of productivity must be balanced against the potential effects of such an exchange on the quality of the corporate information base; a resource which is increasingly recognized as one of the principal assets of any business.

• END