When Not to Use a Data Base

by George Schussel

Data base technology is too complicated and expensive a tool to use for all applications.

Without any question, the use of data bases and data base management systems (DBMS) has been a primary trend in dp shops since 1970. Along with the technologies of minicomputers and communications, data base has become one of the three major differences separating the decade of the '70s from the '60s.

There also is no question that the use of data bases has come a long way in the last five years. As recently as 1970 there were about 100 users of DBMS in the U.S. In 1975 one can add up the number of installed customers of major packages such as IMS, TOTAL, IDMS, ADABAS, SYSTEM 2000, DMS1100, and IDS, throw in home-grown DBMS, and conclude there are around 3,000 users of data bases and associated DBMS in the U.S.

Not only is there tremendous interest in the subject, but some people are comparing the impact of data base on dp in the '70s as being analogous to that of COBOL in the '60s. At the start of each decade, each respective concept was only an idea where by the end of the '60s, COBOL had achieved the status of de facto standard within most shops, and data base well might by the end of the '70s.

Truly, there's a bandwagon effect. Who wants to be left behind with obsolete technology? Well, it seems that many are finding out that tape processing on sequential files is not obsolete, and will remain an important mainstay of the dp shop for many years to come. While it is certain that the data base approach will become a standard in most shops, it is equally certain that many people have made the data base decision for the wrong reasons; the remainder of this article points out some good reasons why you should consider *not* going to data bases at this point in time.

Data independence

The primary rationale for the data base approach is its facility for data independence—in other words, the divorcing of the description or definition of data from any individual program, thus allowing multiple users and programs to access the same data files without undue difficulty. Since that is the primary reason for installing a data base management system in the first place, then the converse should also be true: for anyone who does not need data independence, there certainly is less rationale for going to data base.

It is easy to think of applications where it is desirable for many programs to have access to the same data. In the insurance industry, for example, systems and the data files to support those systems tend to be highly integrated. The data base approach offers a wealth of advantages by eliminating redundancy and/or awkward construction of systems to achieve the goal of multiple uses of the same data.

On the other hand, it is more difficult to think of an environment where there is little or no desire for data to be shared among users. However, consider for example a service bureau environment in which computer time is sold to many small users who access their own individual files for research and problem solving. If the individual users do not want to share information in their files, there is little need for the data base approach.

Sequential processing

Since tape drives were delivered on the UNIVAC I and on practically every general purpose medium-to-large scale computer built since that time, most data files have been built around the use of tape as a basis for organizing information. Tape is inexpensive, does not have to be kept on-line, can be passed at very high speed, and is ideally suited to handle large volumes of data.

However, the only practical way to handle information on tape is to store it sequentially. Sequential storage and accessing tends to be pretty good for some applications, but extremely bad for others. Tape is good when you have a relatively high hit ratio. Various studies (see for example William R. Charles, "Some Systems Shouldn't Use Chained File Techniques," *Data Management*, Sept. 1973, pp. 33-37) have shown that whenever you are passing a file with a hit ratio of more than 4-6% of the records, then a sequential based organization is probably the most efficient. (If you are hitting on a very small percentage of the records, some sort of "random" access methodology on disc is to be preferred.)

If the primary consideration is speed of processing, and the hit ratio is over 4-6% consistently, one should then want to stay with tape files and sequential organizations. As the hit ratio rises, the advantage of tape over disc goes up dramatically.

Another reason for staying with tape-based processing is when there is no current or foreseeable need for on-line access to your files.

Tight security requirements

Most DBMS come with some type of privacy or security mechanisms. CODASYL systems use the subschema approach to implement data security, while other systems use passwords or other lockout mechanisms.

Nevertheless the most secure data is located in a locked safe and only mounted when runs require that data. If you need this much security in handling your data, then the multiple user and data independence advantages of the data base approach do not buy you much. Most DBMS have privacy mechanisms that are entirely satisfactory for the normal business dp environment including manufacturing, insurance, banking, distribution, etc. However, if you're the CIA, think twice before you install a data base system!

Can't afford more hardware

There is no question that the data base approach will put a heavier load on your computer than was previously there. This becomes obvious when you measure cpu utilization and main

(Continued on page 91)

WHEN NOT TO USE A DATA BASE

memory requirements in shops that have installed DBMS. For example, it is not unusual to find an IMS shop with 500KB of real main memory devoted to IMS alone. Although IMS requires more real main memory than other popularly installed DBMS, all such packages will consume some real main memory.

Also, because the functions of a data base management system are central and housekeeping in nature, it occupies resources in the computer comparable to those of the operating system. As a result, even on machines like the larger powerful 370s, cpu utilization goes up substantially after installation of a DBMS.

In addition, the data base approach strongly implies the reorientation of your data files from tape to disc. Disc is substantially more expensive for storage than is tape; and, if you're putting all of your files on-line to a DBMS, you'll need to upgrade disc capacity substantially.

Hardware is not too reliable

High reliability of hardware and software is absolutely crucial to the successful installation of a DBMS. One nice feature of tape-oriented processing is that as an automatic byproduct of normal processing modes, backups are created. When one updates a tape file, he does not write over that tape file, but creates a new masterfile. The backup then is the previous generation tape masterfile plus the transaction file which was processed against it. Thus whether you want it or not, you have an efficient backup scheme when you're processing in a tape-oriented environment.

This is not the case in disc-oriented data base processing since you are writing over the individual records in your data base as you are updating. While a number of vendors have come up with a successful warm restart, roll back, and roll forward recovery procedures for their data bases, implementation of these facilities does require analysis and thought. A failure in any of these facilities can result in substantially more problems for the user than he would likely encounter in a tape environment.

Current investment

When the IBM 360 was announced, emulation of programs written for older machines was a key selling point. However, once the realities of the third generation became known, it was clear that emulation carried a heavy price—substantially more hardware! If one wanted to control hardware budgets and at the same time have maintainable programs, the right answer for many shops was simply to close down the programming section to any new requests for one or two years until all of the old programs were rewritten into COBOL or another third-generation language. Where this happened, and the result was well-documented, well-running, easily maintainable systems, then it certainly seems that any proposal to move to another dp concept would have to be suspect.

In any move to data base, a primary candidate should always be that system whose time has come to be replaced because it has been patched to death. Once you've made a decision to upgrade and replace a system anyway, then why not do it using the latest technology—data base? On the other hand, if you've already replaced the out-of-date technology and you're living in a satisfactory environment, then the advantages of the data base approach had better be substantial for justification.

Not yet a standard

Even though CODASYL, the same group which was largely responsible for the development and standardization of the COBOL language, has been actively involved in the data base approach since the late '60s, the urgency of developing a standard for DBMS has not yet taken on the stringency of the COBOL standard in the '60s.

Although over one half of the different DBMS being marketed are exact or close implementations of the current CODASYL Data Base Task Group recommendations, this number is a little deceiving, since most of these CODASYL implementations are relatively recently announced packages, and also are for equipment other than IBM. (The only widely used DBMS for IBM equipment, which is an implementation of the CODASYL specifications, is Cullinane Corp.'s IDMS.) Because the older DBMS were written before the CODASYL specifications came out, the vast majority of all DBMS installations, with systems such as IMS, TOTAL, and IDS, are not CODASYL implementations.

An important question, therefore, is will there ever be national or international standards? Does the American National Standards Institute have any plans for adopting the CODASYL standards? And, if ANSI does adopt this standard, will the various vendors move to it?

If there is a national standard, it will probably be promulgated by ANSI and be either exactly the CODASYL standard, or a variant of it. If there is to be no standard, then there is no reason to hold back from implementing a DBMS. But a move to a standard DBMS from a non-standard one will involve a conversion comparable to that from DOS to OS.

The first move into any new technology necessitates many one-time costs which are very difficult to recover on the first application. With data base, the first application will require either purchase or lease of software not previously used at the installation. The popular DBMS packages license for anywhere from around $30,000 to $160,000. In addition there will probably be a need for additional hardware—especially in main memory and disc, as discussed above.

There will also be the necessary time investment in experienced personnel to acquaint them with the data base approach. (Two to four weeks for competent coding of simple reports, and from four to eight months for a good overall knowledge of the DBMS). Initially the team will have to analyze packages and subsequently learn the techniques of building a data base system with all of its associated technologies such as data dictionaries, data base administration facilities, etc.

Given the significance of these investments, it cannot often be argued that the first data base application will save your company any money. Almost always the payoff comes in later applications as advantage is taken of the elimination of data redundancy, and as the better control that is inherent in the data base approach comes about. If the budget is ominously tight, however, second thought should be given before moving into a data base.

One of the cost aspects mentioned above, the training of personnel, can be expanded into a whole problem area of its own—the lack of adequate numbers of trained personnel. There currently is a large imbalance between a small supply and a large demand for trained personnel in data base technology. As long as the supply is substantially less than the demand, then companies will have to train their own personnel instead of going to the marketplace to hire experienced data base personnel. This is costly since training a senior systems analyst to become a data base administrator can consume six to nine months, plus the associated salary costs of this time. Concomitantly, the probability of losing trained individuals will also remain high as long as the supply and demand for personnel is severely out of balance.

Files being too large is one of the most fundamental reasons for avoiding the data base approach. This cost is also one of the easiest to calculate for your company. There is nothing inherent in data base about either the hardware or the software that limits applications to small files. On the other hand, as the size of masterfiles for the data base grows larger, the implemen-

(Continued on page 98)

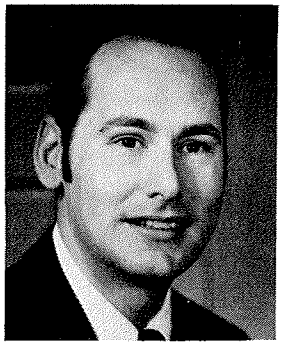
tation problems (schema generation, security, recovery, etc.) grow at least as fast as the data base size.

Most data bases that have been implemented in the U.S. have been in the range of under 100 million characters of raw information. Even today a data base of over 1 billion characters of information is considered very large. There are no more than a handful of such data bases that have been implemented.

In a typical data base environment, the disc overhead factor ranges from 100-300%. This means that 1 billion characters of raw data are going to result in a storage requirement of up to three times that much space—roughly equivalent to between 20 and 40 IBM 3330 devices. Can you convince management to have that amount of disc onsite? If not, and multiple data bases (without dynamic interaction) are not a good solution, then the data base approach would not be appropriate.

The insurance industry, for example, has moved slowly and surely, rather than by leaping, into the data base environment, and primarily because insurance files often run to over 1 billion characters. The associated hardware costs required to implement this data on disc has required careful analysis and slower than desired progress into the data base environment.

The data base approach is relevant and essential to data processing development during the remainder of the '70s and through the '80s. At my company, American Mutual Liability Insurance Co., we are moving slowly but surely into the data base environment. Careful consideration and analysis of the relative advantages and disadvantages of the data base approach is needed before hopping onto the data base bandwagon. ❄



Dr. Schussel is vice president of American Mutual Liability Insurance Co. where he is responsible for the company's dp activities. He received his doctorate from the Harvard Business School and authored a number of articles on data base and related topics.