Data base—An emerging organizational function

by RICHARD L. NOLAN

Harvard University
Boston, Massachusetts

A vignette of increasing frequency is for a manager with a problem to arrive in the EDP manager's office and ask for information required to solve his problem. The information required almost always cuts across computer applications and files of several functional areas. In the majority of the installations, the EDP manager is forced to flatly turn down the manager's request. He either cannot respond because the needed data is hopelessly locked up in existing poorly documented computer files, or because responding would throw the normal computer processing into upheaval.

In a minority, but growing number of other installations, however, the EDP manager is increasingly able to effectively respond to the manager's ad hoc request for information. These EDP managers have been implementing a data base approach through employment of data base software. At this point, extreme caution of interpretation and extrapolation is warranted. Data base or DB rings too much like the illusive Management Information System, or MIS, of the past and the present. It clearly does have some of the same characteristics, but as I will demonstrate in this paper, it is founded on a much more sound footing. I will also provide evidence that the forces are in irreversible motion. But similar to other major computer innovations, the early implementation is expensive and inhibited with organizational change problems. These problems should not be given short shrift, nor should the interpretation of their significance be allowed to overshadow the direction that has been set in motion by the technology.

AN HISTORICAL STAGE PERSPECTIVE

A recent research study revealed that the EDP budget for a number of companies, when plotted overtime from initial investment to mature operation, forms an S-shaped curve. An analysis of the events associated with the S-shaped EDP budget curve led to the formation of a stage hypothesis of EDP growth. Exhibit I shows the S-shaped EDP budget curve pattern and the three growth processes that must be dealt with as an EDP function matures: (1) growth in EDP management techniques, (2) growth in specialization of personnel, and (3) growth in computer applications.

Because of the rapid growth of computer technology, management of data has developed haphazardly and in a laggard fashion over the years. A general approach of data management has emerged only very recently, and, consequently, applications have developed discretely from one another in an unintegrated and wasteful fashion. Further, each increase in the complexity and capabilities of computers has brought new generations of applications, but these applications still, for the most part, have been specialized in nature, designed for a specific operational use or for a specialized staff function.

Hence management of data has continued to develop in fragmented fashion and at rather low organizational level—at a subdepartmental or substaff level.

Today, upper levels of management are seeking information that can be generated only from properly structured, company-wide pools that include data from the narrower applications located further down in the organizational hierarchy. That is, management information today requires that a company have a data base which can be used in conjunction with broad-range programs, to generate information on a broader and more comprehensive scale than the single, isolated applications of the past could usually do.

Notwithstanding the new demands, tradition is still strong; indeed, it has barely been challenged. Exhibit II represents the traditional way of doing things—collecting and coding data for specific programs and thereby gluing them more or less permanently and exclusively to those programs. In retrospect, this approach has had three significant disadvantages.

Files and records have tended to become redundant

Suppose Company X originally had only a single, computer-based system—say, for accounts receivable—which is represented in Exhibit II as Program I.

Program I has three data files: A, B, and C. File A contains customer records, each consisting of data elements a and b; a might be the customer name and b his outstanding balance. Files B and C contain other data elements needed for the accounts receivable program.

Assume that now the company wishes to implement a second program—Program II, as illustrated in Exhibit II—with Files D, E, and F, comprehending elements a, b, c, d, f, and g. Note that the company already has all these elements, except g, on file for Program I. In all probability, however,
its programmers coded Files A and B (including all the elements a, b, c, and d) expressly for Program I, and hence cannot now use A and B intact for Program II. Thus the programmers have to make a choice:

(a) They can recode A and B so that these files can be used by either Program I or Program II. But this would mean rewriting Program I to take account of the recoding.

(b) Alternatively, they can build two “new” files, consisting of data from A and B but coded for the special convenience of Program II.

In the past, when faced with this kind of choice, an EDP department has usually just gone ahead and constructed the two “new” files. Going back over Program I ordinarily seems like too much trouble, so making up the new files seems the easiest way out. It is—in the short run.

But in the long run, as the exhibit shows, Company X might easily find itself creating more and more quasi-duplicate files as it adds new programs. For example, it will need two new versions of File B for Programs II and IV—that is, Files E and K. It will need three new versions of File A for Programs II, III, and IV—that is, Files D, G, and J. It will need a new version of File I for Program IV—that is, File L. And so on. The redundancy of data is obvious. In just this little, highly simplified example, 7 out of 12 (58 percent) of the data elements of the files are redundant.

Initially, redundancy does not cause a great deal of trouble. As soon as pieces of data must be updated, however, it does cause a great deal of trouble. In an EDP department of any size, it is virtually impossible to update all the redundant files and reports in systematic and synchronized fashion. Consider what must happen if Company X adds a customer: it must update A, B, D, G, and J, and that would only be the beginning.

Once files, records, and reports have begun to overlap and updating becomes a serious chore, updating procedures begin to sag of their own weight and different parts of the organization begin to receive inconsistent reports generated from files that are in various states of disrepair. In one large company, the inconsistencies between sales reports at the division level and sales reports at the branch level were so extreme that the salesmen began to keep very elaborate manual sales records. These two sets of reports were, in fact, generated in large part from redundant files that were updated at different times.

These particular inconsistencies resulted from a mere difference of organizational level—that is, the divisional versus the branch level. Severe redundancy problems can arise even more easily when reports from one function must be meshed with reports from another function. For example, there is absolutely no reason to expect that a company’s inventory-control report will jibe with its accounting report unless the updating disciplines for the files of both functions are synchronized with each other.

Even slight variations in the data used for the two functional reports can cause glaring inconsistencies:

In a large retail chain whose applications had developed in the traditional fashion, the needs of the business forced management to request the integration of a number of different functional programs and systems. With great effort, the job was done. However, it was done in such a way that many quasi-duplicate files were created and many separate, but essentially similar, programs were patched together. The company suddenly found itself spending 90 percent of its programming man-hours just keeping the programs running in concert and the files up-to-date.

At the very least, redundancy, spells confusion and expense for any sizable operation. Perhaps its worst feature is
that the longer a company follows the traditional pattern and keeps adding new programs and redundant files of data, coded specifically and exclusively for those programs, the greater the task it must face when it finally assembles all its data in a single pool, so structured and coded that new programs can be run without extensive recollection or recoding of data.

The traditional approach undercuts or aborts the advances of computer technology

Originally, the relatively high cost of on-line storage was a main factor that induced companies to delimit the scope of programming and therewith the amount of data needed during any given run. In effect, this reinforced the practice of creating and maintaining separate files for each application in the company's portfolio—companies tended to store no more data than were needed for the run at hand.

Today, however, many companies that have followed the traditional route, but have acquired up-to-date on-line storage systems, find they have the capacity to keep relatively huge amounts of data alive in the system. But their data are still organized and coded along first-generation computer lines—that is, by specific programs. From a rational viewpoint, this is as awkward, expensive, and absurd as keeping modern accounting records wholly in Roman numerals.

The traditional approach obstructs upper management's growing demands for applications that require a data base

A review of the evolution of computer-based applications runs as follows. First, the computer was first used to replace existing manual functions, primarily within the accounting function. Next came the integration of computer-based systems within and between functional areas. Now cross-functional/interlevel systems are being developed to serve middle and upper-middle management; or, to put it another way, management is now demanding the benefits of computer innovations. The redundancies and inefficiencies that result from the traditional approach to the management of data become so signal and so extensive that applications can be adequate only if they are developed in such a manner that specific programs are separate from the data.

THE DATA BASE CONCEPT

As Exhibit III shows, the data base concept is to structure a company's computer-readable data into a single pool, which is used to run both routine programs and programs written in response to ad hoc requests. Note that no files appear in this exhibit. The base of data elements and their structure supplant the specific files. Note also that two additional software systems are in evidence here which were not in evidence in Exhibit II:

(a) The data base interface system enables a specialist data base programmer to organize and structure the data elements in a manner that minimizes or eliminates redundancy and optimizes the economic costs of data storage and accessibility.

(b) The interface system for special programming includes a high-level programming language especially designed for manipulating data elements contained in the data base, solving problems, and producing reports. To write ad hoc programs, the programmer works successively, through the interface for special applications and the general interface system to the data base itself.

Comparing Exhibits II and III, one can see an immense contrast between the traditional concept and the data base concept, both theoretical and practical.

Since much of the computer technology necessary to implement the data base concept exists and the rest of the technology is being developed rapidly, a strong case for adopting the data base approach can now be made. Yet, in operational terms, the concept is still novel. To what degree is it being used? What are the issues and problems involved in implementing it? By what strategies can a company work toward a data base? And what benefits can we realistically expect from it?

AN INTERVIEW STUDY

To answer these questions I administered a pattern interview to the data-processing managers of ten companies in six diverse industries. The questions permitted unrestricted
responses, and hence the information these managers provided is not as clear-cut as one might wish. However, it is informative. The opinions expressed varied considerably among the EDP managers. By and large, a given manager's opinions reflected the particular stage his company had reached in the evolutionary progression toward full use of the database concept.

First of all, I found a certain amount of confusion about what "database" means. My open-ended question, "What is the database in your company?" usually brought first a puzzled expression to the manager's face, and then a request for clarification. I answered that I wanted a statement on how he views his company's database, if, indeed, he views it at all.

Responses ranged all over the lot. Some managers included all the computer-readable data in their company. Others defined the base more narrowly—for example, including only the random-access disc files used for routine reporting and analysis.

The common thread in the responses was "computer-readable." Since all the interviewees were data-processing managers, this common thread is not surprising. But, obviously, the great majority of an organization's data are non-computer-readable; they are maintained in file cabinets as well as in the minds of management.

Although more and more data are being put into computer-readable form, as the technology improves and makes more sophisticated computer-based applications both feasible and economic, much of the literature on data bases falsely assumes that companies have already translated all the data needed for these applications into machine-readable terms. This simply has not yet happened—indeed, most companies have not even begun to collect the data needed for these applications, in machine-readable form or otherwise.

In general, the more advanced a company's use of the database approach, the less naive and more realistic the manager's definition of what the base ought to contain—for example, 'shared random-access files used for periodic production programs and ad hoc management requests.' Such a definition reflects the two key characteristics of the database: (a) sharing data between programs, and (b) structuring data so that ad hoc management requests can be served.

One data-processing manager articulated the criterion of responsiveness to ad hoc management requests especially well. He said that his company will realize the database when he has incorporated the technology that will permit him to respond to any reasonable request by management for reporting or analysis within one day, and without undue degradation of his continuing data processing. He further described a reasonable request as one that draws on existing computer-readable data.

Responses to my questions on the structure and integration of computer-readable data grouped into three categories. On one end of the spectrum were the companies that structured their computer data on the basis of individual applications, with only limited cross-functional integration among applications—that is, sharing data between such functional applications as manufacturing and accounting. In the middle of the spectrum were the companies that had aggressively designed applications for cross-functional data sharing. On the other end of the spectrum were two companies that not only had achieved a high degree of cross-functional integration of their database, but also had begun to achieve an interlevel integration—that is, an integration of their database for managerial reporting and analysis with their operational data bases.

Both of these companies used commercial database software packages. They used one software package for data organization (the General Interface System of Exhibit III), and a different software package to produce ad hoc management reports and analyses (the Interface System for Special Applications in Exhibit III).

Both data-processing managers were reasonably satisfied with the commercial software they were using. Nevertheless, they both commented that even the most sophisticated database software commercially available did not incorporate the more advanced data-structure methods. Such methods coordinate theoretical data structures (for example, things resembling immense decision trees) with the access constraints of physical storage devices, such as rotating magnetic discs. Suffice it to say that data organization is extremely complex and technical.

It is so complex, in fact, that one is virtually forced into using commercial software. One of the data-processing managers stated that structure technology is so complex today that he could not possibly support an in-house effort to develop the software. The other manager had initially hoped to develop his own database software, but, after a preliminary investigation of the costs and problems, he decided to acquire a commercially available package.

However, this complexity ultimately derives from the nature of the key tasks for which top management wants the data base to be used. If upper management focuses on key tasks that embrace all the company's data and require very extensive vertical and horizontal integration of reports and analyses, the job of organizing the data base is tougher than when the key tasks embrace only a part of the data and require less than the complete integration of all functions.

The EDP managers interviewed expressed concern over these major organizational issues associated with the database:

(a) Acquiring personnel that can handle its technical aspects.
(b) Funding and developing suitable charge-out systems to support it.
(c) Setting and enforcing company-wide standards.
(d) Using the data resource to best advantage.

The major associated technical issues for which they expressed concern were these:

(a) Converting data to database form.
(b) Providing appropriate software for the interfaces.
Data Base—An Emerging Organizational Function

(c) Designing a data base which will permit ad hoc responsiveness without degrading normal computer processing.

(d) Building in reliability and the ability to reconstruct lost data.

Both the organizational issues and the technical issues were generally felt to be of such magnitude that aggressive action on implementing the data base concept fully was not warranted at the time. The consensus was that the concept is sound, but that much more needs to be done administratively before it can be effectively realized in practice.

MANAGEMENT ACTION4

Use of the data base concept is the next natural milestone in the evolution of EDP applications. It embraces the specialization of EDP functions; it allows management real flexibility in satisfying its need for information; and it permits companies to view and use their data as a real resource. Yet caution and patience are advised in pursuing the concept. What should managers do to deal with this push-and-pull condition?

Set up a data administration function

The issue is when to set up a data administration function, rather than whether to have such a function. Ultimately, an administrator will be needed to implement the data base concept, anyway. For those companies currently without such a position, an administrative structure is needed for formulating a data base implementation plan, and establishing data base standards, controls, and access procedures. At a minimum, a data base specialist should be acquired now to provide decision-making guidance for the EDP manager and steering committee. This person can also provide guidance in evaluating and selecting appropriate software.

Incorporate data base technology into the computer system

The hardware technology, as well as the software technology, for data bases has matured to the point that the data base concept can be both feasible and cost-effective for many organizations. While the company will not be noticeably hurt in the short run by ignoring data base technology, it will in the longer run.

Also, the data base concept cannot be implemented overnight. If a company begins to plan and act now, it can assimilate even drastic technological improvements into its existing systems in an orderly fashion.

To incorporate the technology that will permit data base operations, an organization must identify its key computer-based systems and restructure them (a) to remove redundancy, and (b) to facilitate their use by higher levels of management. For the present, companies must probably acquire commercial software for structuring data and responding to management requests for ad hoc analyses and reports.

Think of data as a resource

For the longer term, management should begin to think of data as a basic resource. It should accept this idea as a natural consequence of functional specialization of the general management function. Since the data-resource concept is closely associated with a fast-moving computer technology, management should expect to see the movement toward specialized data-management activities proceed at a faster rate than, say, specializations in the human resource function.

In retrospect, the curve shown in Exhibit I takes the same shape as the generic learning curve. In large part, I think that the curve has been driven by developments in hardware technology in the second and third generation computer systems. Now, however, the advancements seem to be taking place more in software than in hardware. The breakthrough most likely to start off another S-shaped EDP budget curve is data base technology. In dealing with data base technology, caution is in order and the painful lessons of the past should not be cast aside with the first blushes of enthusiasm in perceiving the potential of the concept.

REFERENCES

4. Ibid., pp. 113-114.