Management Services: A Magazine of Planning, Systems, and Controls

Volume 6 | Number 2 Article 6

3-1969

Expanding the Uses of the Bill of Materials Processor

Jonathan Bayliss

The bill of materials processor, a software package offered by many computer manufacturers, fills the obvious uses for which it was designed. But, beyond that, the author's company has applied it in many cases where traditional methods can't work well—

EXPANDING THE USES OF THE BILL OF MATERIALS PROCESSOR

by Jonathan Bayliss
The Gorton Corporation

THE so-called bill of materials processor, a computer software package offered by IBM, Honeywell, and other computer manufacturers, is a tremendously powerful tool.

The bill of materials processor is a package of programs that will create, maintain, and retrieve in various forms the bills of materials of a manufacturer, particularly those of a fabrication and assembly manufacturer. With suitable customizing it will provide three kinds of explosions — single-level, indented, and summarized. It also will give the same three kinds of implosions. In other words, the bill of materials processor will retrieve, extend, and organize a manufacturer's material requirements, past or future, six different ways.

All this is impressive enough. But it is only the beginning of what the bill of materials processor can be made to do by a skillful user.

More than a package

Actually the bill of materials processor is not a programing package in the ordinary sense of

the word but a basic piece of software of far more general significance. It is a method of file organization; it is a package system that can be applied to many requirements for which the usual methods of file organization are unsatisfactory.

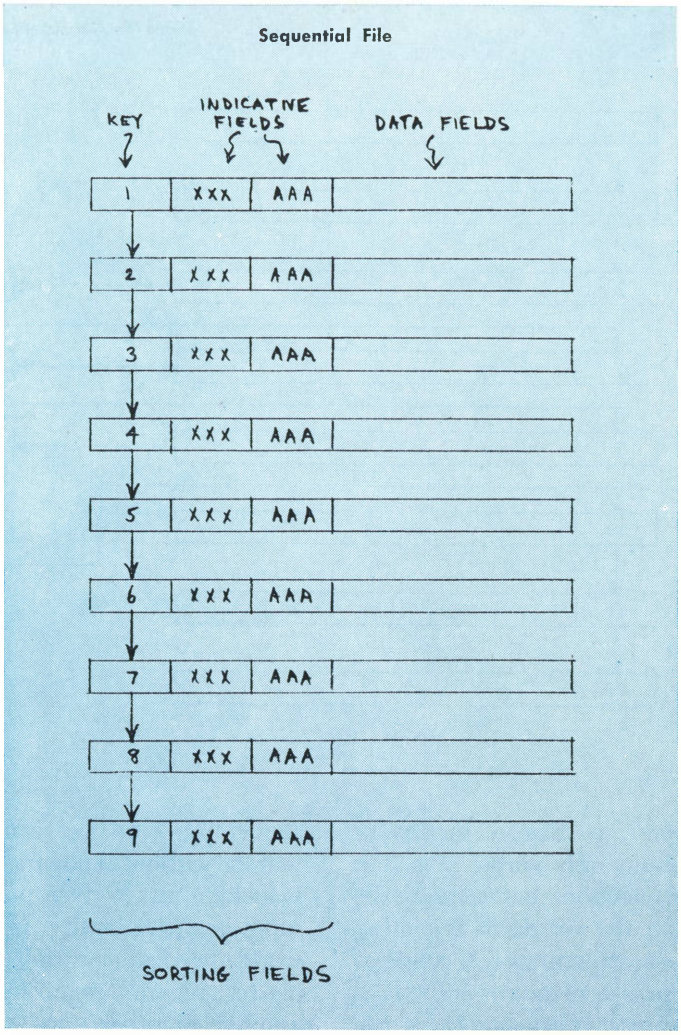
At Gorton (a frozen seafood processor with national distribution and international resources, which recently merged into General Mills) we use the bill of materials processor for operational cost control and logistical control, and we plan to use it soon for property administration. These three applications are described in some detail in this article. They do not begin, however, to exhaust the possibilities of this versatile tool. Once the principles of its use are understood, it can be adapted by an ingenious systems man to a wide variety of applications.

File organization

On our IBM System/360 Model 30 we basically use three different kinds of disk files:

We use sequential files (Figure 1 on this page) generally for high volumes of data with very high growth rates, when the records have coded connections to each other, when mass high-speed processing is required, and when there is little need for random access. With such files we rely on selection and sorting techniques to a very high degree, and it is in this manner that we establish the relationship between records. Our general data base for sales statistics, accounting, and inventory is a multi-pack sequential file.

FIGURE 1



Random access files

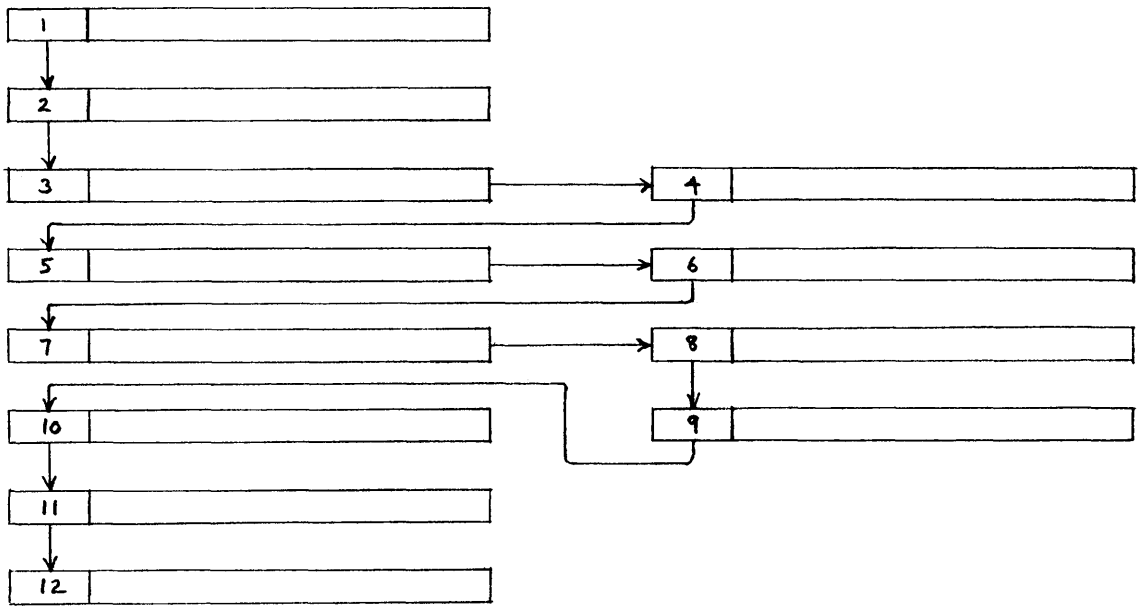
We use index sequential file organization (Figure 2 on page 46) when the records are sometimes used in sequence but often accessed randomly. In this case the relationship between records depends primarily upon the key. These files are used primarily for master reference and are almost always on line for purposes of supplying constant information in a nonredundant manner. Master reference data, such as product description and standard costs, appear only once in the system, and the file which contains them is usually running in parallel with one of the other two types of files.

The third type of file is the bill processor (Figure 3 on page 47). It is used for establishing relationships between records which need not be related by key or by other coding characteristics. These relationships may be changed, and so may the data content of the records. This statement will be explained more fully later in the article. For the moment, suffice it to say that this kind of file organization is really an organization of two files—a master file and a structure file. The master file contains constant, semi-constant, or completely variable data pertaining to an entity. The master file is like a dictionary that contains only nouns. The structure file, on the

JONATHAN BAYLISS is director of management services at The Gorton Corporation in Gloucester, Massachusetts. He is a member of the American Production and Inventory Control Society, The Institute of Management Sciences, and the administrative systems committee of the Grocery Manufacturers of America. Mr. Bayliss had previously been a consultant, a sales analyst at The Carter's Ink Company, and an administrative assistant to the sales manager at the Malsbury Manufacturing Company. He attended Harvard College and was graduated from The University of California (Berkeley), where he was elected to Phi Beta Kappa.

FIGURE 2

Index Sequential File



other hand, is like a dictionary that contains only verbs, in that it shows connections between, or the actions of, the words in the other file. These structural relationships may be semi-constant or fully variable, and this characteristic is the key advantage of this kind of chained file.

The three types of files are compared in Figure 4 on page 48. The sequential file is like a straight line whose points can be rearranged to a limited degree. The index sequential file resembles a curved line that can be followed in one direction only. The processor files, however, are like a network. It is possible to travel in either direction and to get from any one point to any other point. For this reason we often call the "bill processor" a network file.

Network files

With a network file—or two files or even three if it is desirable to add a third dimension—the user can reverse or alter his search (unlike with a sequential file) without sorting, and (unlike with an index sequential file) he can make a search without knowing what he is looking for. A given point, which represents an entity in the real world, need appear only once as a master record, regardless of the number of other entities it is related to. Connecting lines can be changed without changing the identification of the entities involved. In other words, the file provides the tools of construction; it offers a way of putting building blocks together like a limited number of atoms which can make up an astronomical number of different molecules. There are converging networks and diverging networks and hierarchical networks —all of which can be modeled with this kind of file. (See Figure 5 on page 49.) And there are also networks that are composites of these.

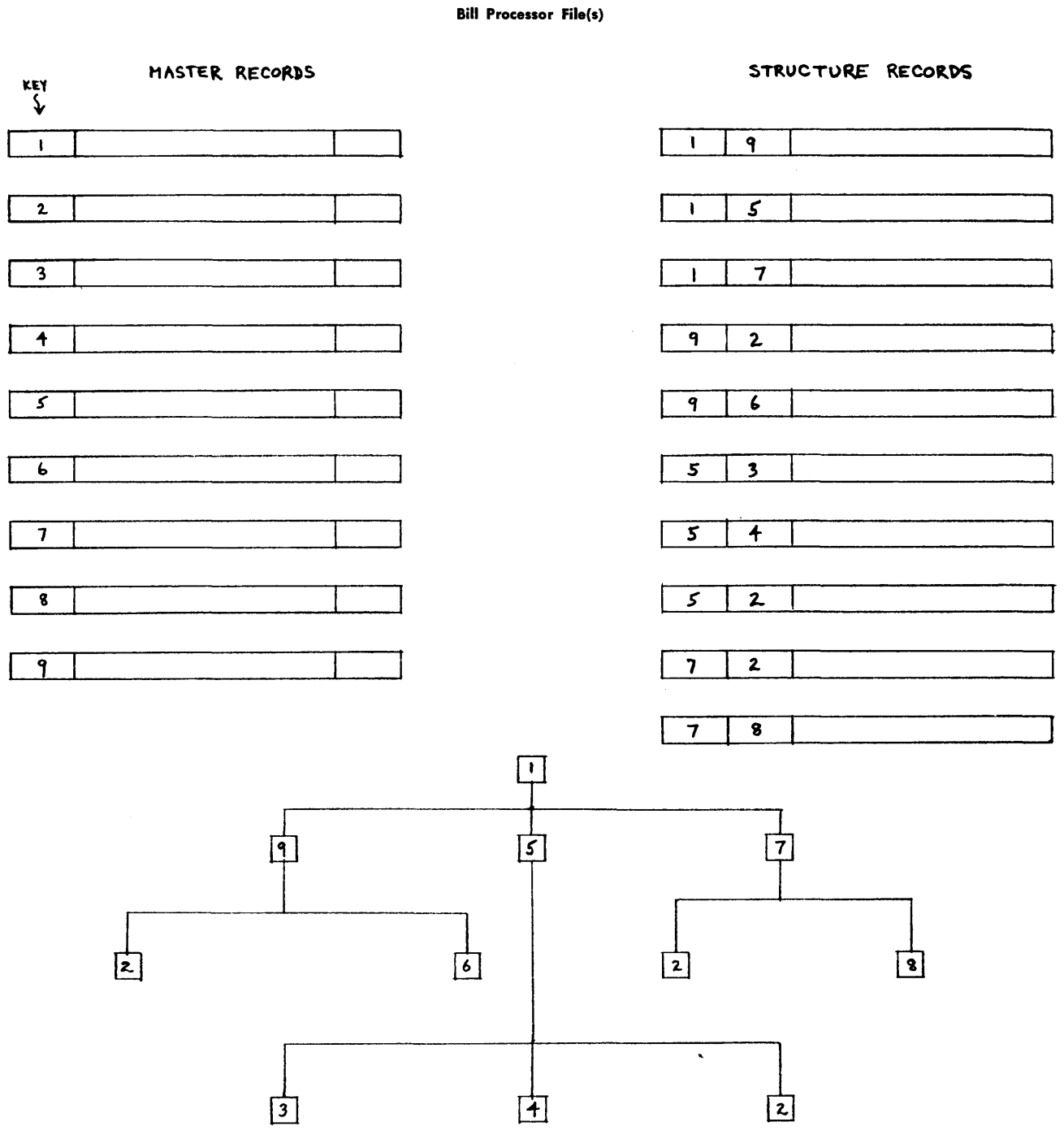
The thing that makes a network file much more powerful than it may have seemed from the discussion so far is the possible uses of the structure records, which are represented by the lines between the points in the diagrams. In the earlier figures most of the records were left blank. In Figure 6 on page 50 these fields are filled in. The master records can have as many fields as the user wishes, and these fields can each be updated in the course of processing—each according to its own programed rules. In other words, variable quantities can be assigned to points in the network, and each point can actually be a cluster of smaller points.

Field as valve

More important, however, is what can be done with the fields in the structure records. One field (S1 in Figure 6) would normally be a coefficient. It could be thought of as a valve that is set at any point between fully on and fully off. The several valves through which the data flow—let's say, on the way down from the top—thus determine the proportion of the flow that takes one branch or the other. When there is only one connecting record, naturally, 100 per cent has to flow in that one direction only.

The second field, shown in the structure record (C1), which again may be made up of several

FIGURE 3



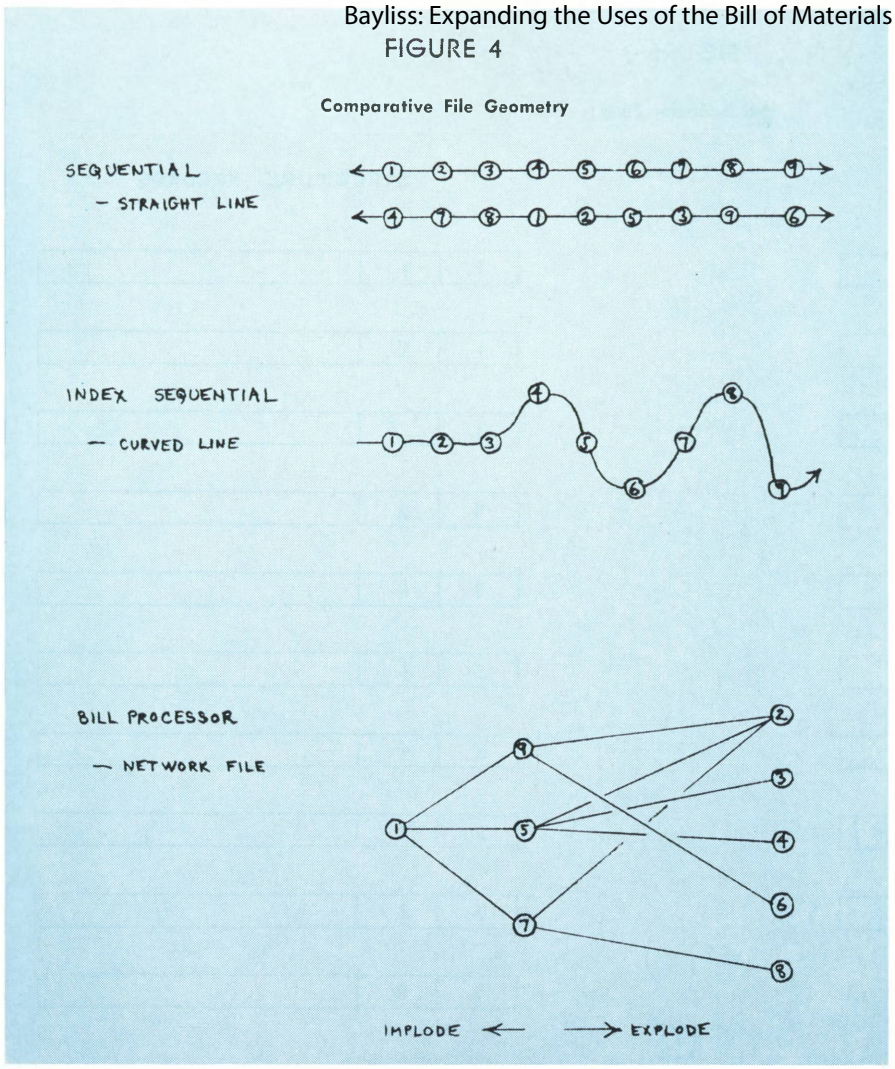
sub-fields, may contain control data of its own. This data also may be updated during processing. It may measure the flow through its leg of the system. Or it may shape or modify the information that is flowing through the system according to certain characteristics.

The diagrams of these network types should suggest many applications of the bill of materials processor. A manager who needed a PERT or a CPM program and lacked a specialized package could use the network file to tailor his own planning system. It could be used in library management for cataloging books and retrieving bibliographic cross-references by subject, by author, by title, etc. It could be used for organization charts that show functional and working relationships as well as nominal chains of command. It could be used for complicated cost allocations in an accounting system.

Uses of program

It can be used for almost any purpose that is best served by a structural model when both the structure and the content are variable. It can be used for chaining

FIGURE 4



historical facts in a cumulative fashion so that when reference is made to the present condition of, say, a person in a hospital, it is also possible to retrieve an unlimited number—a variable number—of previous status reports as well as all the connecting events between them, such as medication or surgery. Theorists say that a system can never be fully designed. This is all the more reason to use a network file, which can grow and change as the real world grows and changes.

Bill of operations

At Gorton we are applying network file technique in three ways. One subsystem has been operational for over a year. Another is approaching the first stage of implementation. For the third programing has barely begun.

The first subsystem is our bill of operations, illustrated in Figure 7 on page 51. This includes the conventional bill of materials, but we call it a bill of operations because it includes more than just materials. It also contains, in effect, elements of processing itself—such intangibles as fixed overhead, labor, storage and handling, and other nonphysical components of our finished products. With this subsystem we update our standard costs, usually once every four weeks, by changing the cost of individual components that may be common to many finished products. We thus can very quickly explode the effect of a change in the world commodity market and swiftly adjust our prices to follow the domestic market without distorting our gross margins on sales. Our reaction time has improved by at least a month.

The bill of operations can be used to simulate tentative cost schedules and—once the sales budget is locked in—to project material requirements for the year ahead and cost them out, period by period, to form the basis of the company's expense budgeting. Naturally, we also explode short-term revised production forecasts for a limited number of time periods. These are known as "summary explosions."

"Indented explosions"

"Indented explosions" are used for audit trails or special analyses (such as in the case of anticipated shortages) to give us the cost and quantity of materials and services required for any given quantity of any given finished product.

A "where-used" retrieval listing is made periodically for each component as a reference manual for the purchasing and cost accounting departments.

"Summarized explosions" are the ones often used. They take several forms. In one form, the summary goes down one level only in the product tree, to what we call "cost categories." These are accounts from our chart of accounts. This summary gives us our standard cost absorptions for posting to the books. Other summarized explosions pick up materials only at their lowest level, that is to say, as purchase requirements. Still other summary explosions may be run at an intermediate level, such as an explosion of production department requirements for materials that must be immediately on hand in the warehouse, in the form and at the value of productive line usage.

In the bill of operations subsystem, since each item appears only once, it is possible to tie together rather complicated processes. For example, we are able to combine batch process accounting with production line items when some of

the components are moved in batches at a previous point in time whereas others are used only on the day of final production.

Underlying our subsystem is the concept of "value added" at each step of the way. This network file is so flexible that we can usually introduce new elements of complication, new cost accounts, new processes, new value increments of any kind without changing the programs. We are able, for instance, to use a "plant cost differential" cost category for plugging and equalizing the standards for finished products made at more than one plant, with different labor rates, yields, and overhead charges.

After all this time we still have a lot to learn about how to use this versatile tool. We have a long way to go in bringing our actual cost data collection to the level of sophistication that will do full justice to the bill of operations in measuring variances by individual items.

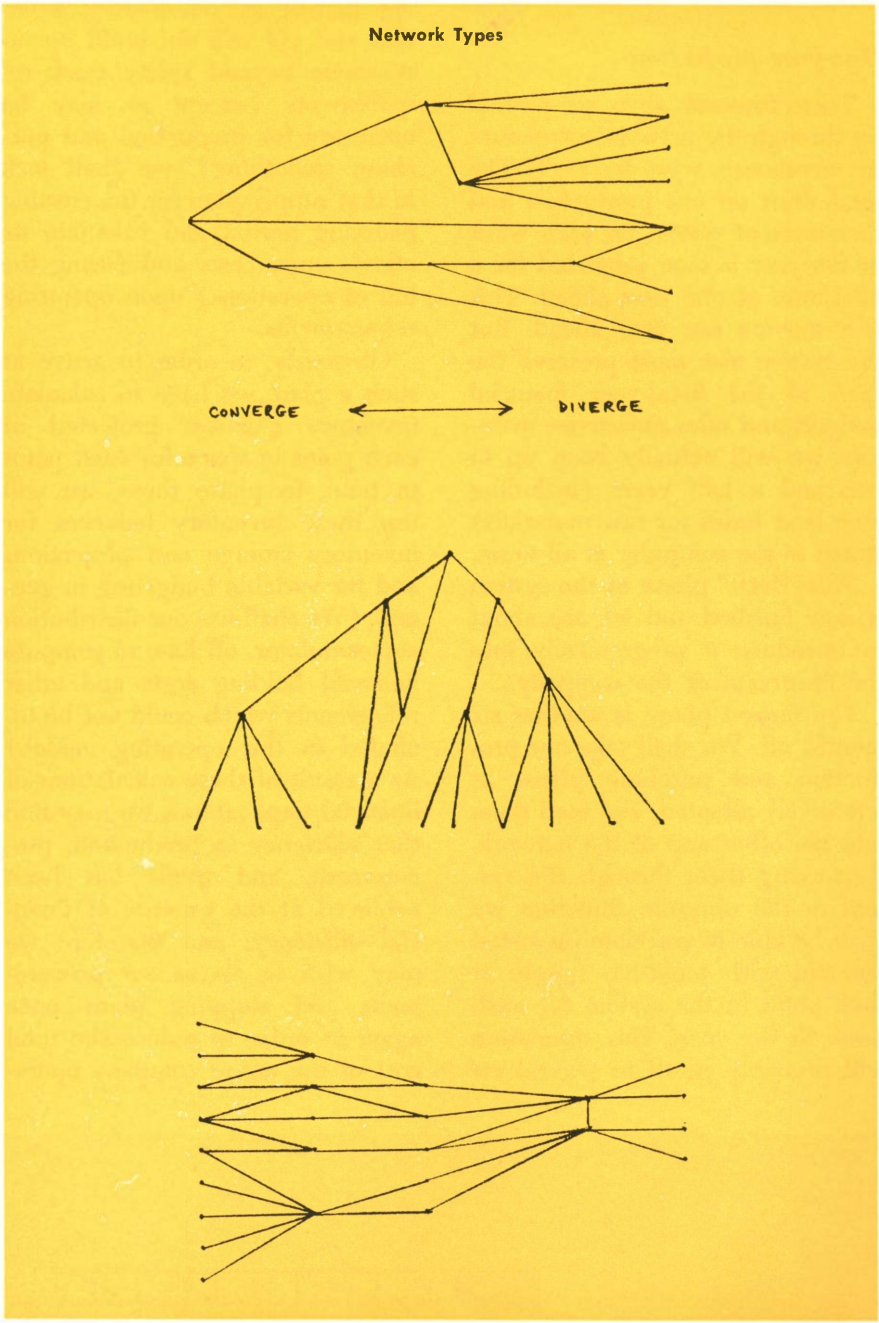
Logistical model

Our second application of network files is the most monstrous. It is huge. It is nothing less than a working model of our entire logistical system (shown in Figure 8 on page 52). It will project sales, production, purchases, and inventories for every stock-keeping unit sold in every one of several hundred markets from any one of a couple of dozen local warehouses or from any one of three distribution warehouses supplied by any of three base warehouses and procured from any production line or vendor.

But we are an inventory management business, and it's worth the effort. And this vast detail—thanks to network files—is not as overwhelming as it appears. Once the model is built in simplified form, it can be expanded in structure step by step, and it can be updated in data content by selective changes as conditions warrant. In other words, our annual fiscal-year budgeting (which centers in sales and logistics) will be in process all year long. Once the initial budget is fixed for the fiscal year, the operations planning system will be used to revise it market by market, item by item, warehouse by warehouse, freight rate by freight rate, etc. This continuously revised budget constitutes the operating plan according to which all operating decisions are made.

Hitherto our operating forecasts, although we have revised them periodically, have not represented a synthesis of local forecasts. The planning has been limited—as most such planning is, everywhere—to the major base and distribution warehouse level. There has been no way to reflect changes in markets, local promotions, loss of major customers, gain of major customers, or special local competitive situations. In other words, we have had no way of formally gathering and reflecting the knowledge of the man in the field. Now, in contrast, the effect of a change in major customers in San Fran-

FIGURE 5



cisco will be felt in the production planning for the following period.

One-year projection

These forecasts, then, are pushed up through the network warehouse by warehouse, week by week. The total draft on our production and procurement system for each week in the year is then projected for a minimum of one year ahead. This is a moving one year ahead. But the system also must preserve the span of the fiscal-year financial budgets and sales objectives; therefore we will actually keep up to two and a half years (including long lead times for raw materials) stored in the computer at all times.

This "draft" phase of the system is now finished and we are about to introduce it progressively into the lifestream of the company.

The second phase is another six months off. We shall take our production and purchase plans, as tentatively adopted, and feed them into the other end of the network. By forcing them through the system in the opposite direction we shall be able to correlate projected demand with tentative supply at each point in the system for each week in the year. This simulation will probably result in several revisions of the supply plan. Quite supply plan is found which will not permit an out-of-stock situation and yet will not build up inventories beyond safety stock requirements (except as may be necessary for production and purchase smoothing), we shall lock in that supply plan for the ensuing planning period and calculate its effects upon costs and (using the bill of operations) upon operating requirements.

Obviously, in order to arrive at such a plan, we have to calculate inventory balances projected at each point in space for each point in time. In phase three, we will use these inventory balances for inventory storage cost projections and for variable budgeting in general. (We shall use our distribution cost simulator, off line, to compute financial holding costs and other refinements which could not be included in this operating model.) As a result of these calculations of financial implications, we may find that efficiency in production, procurement, and traffic has been achieved at the expense of financial efficiency, and therefore we may wish to revise our procurement and shipping plans once again in order to reduce the total cost of the entire company operation. In short, the operation planning system is itself a simulator.

It is also the framework within which all other logistical uses of the computer will fit. For example, as an extension of our present distribution warehouse replenishment calculation, we are about to install the new IBM Inventory Control System, which is a sort of manufacturer's version of IMPACT.* The intrinsic forecasting involved in this subsystem will act as a kind of governor on the long-range extrinsic master forecasting that I have just described. They will couple with each other, and we hope to get the best of both worlds. Also, sooner or later we expect to develop our own production scheduling matrix (probably a simulator of some type) for balancing up production lines in time and relative load for the purpose of achieving the best practical production efficiency consistent with the seasonal variables of both sup-

*IMPACT (Inventory Management Program and Control Techniques) is a programing package for inventory control supplied by IBM. Its operation was described in some detail in an earlier article in MANAGEMENT SERVICES ("An Inventory Control System with Profitable By-Products" by Anthony A. Vallario, January-February '67, p. 31).

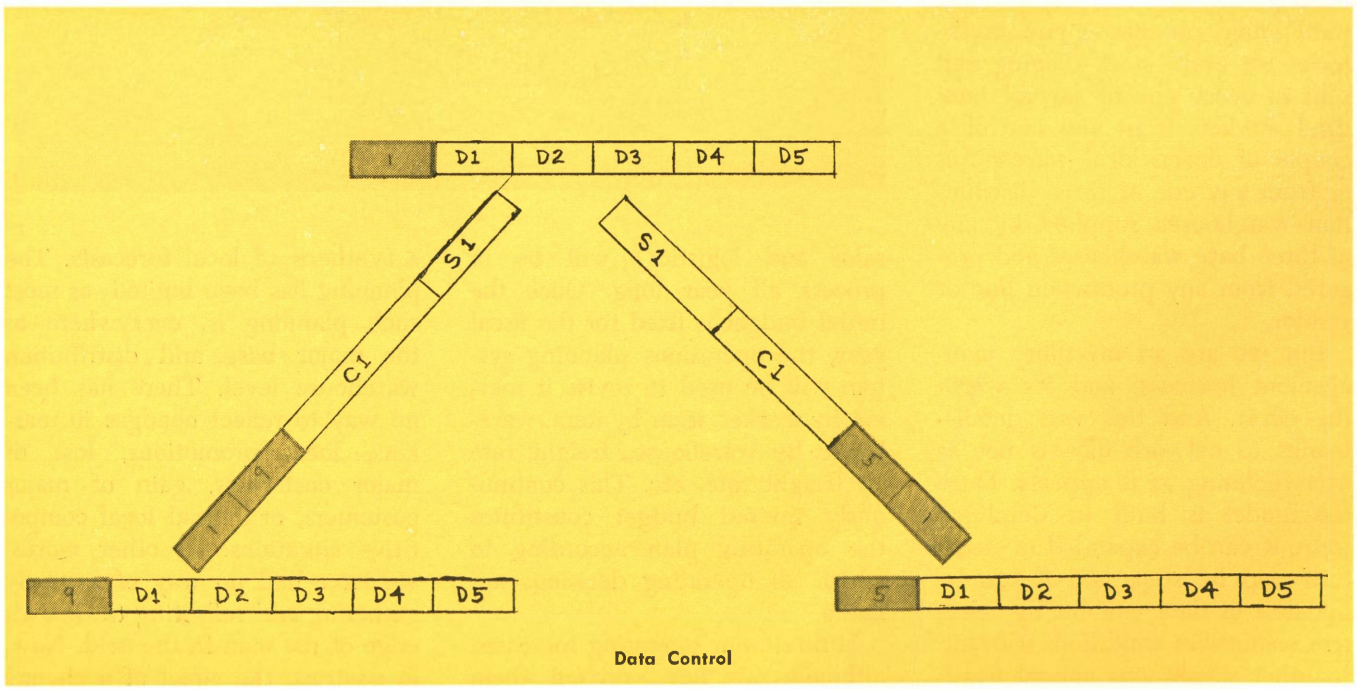
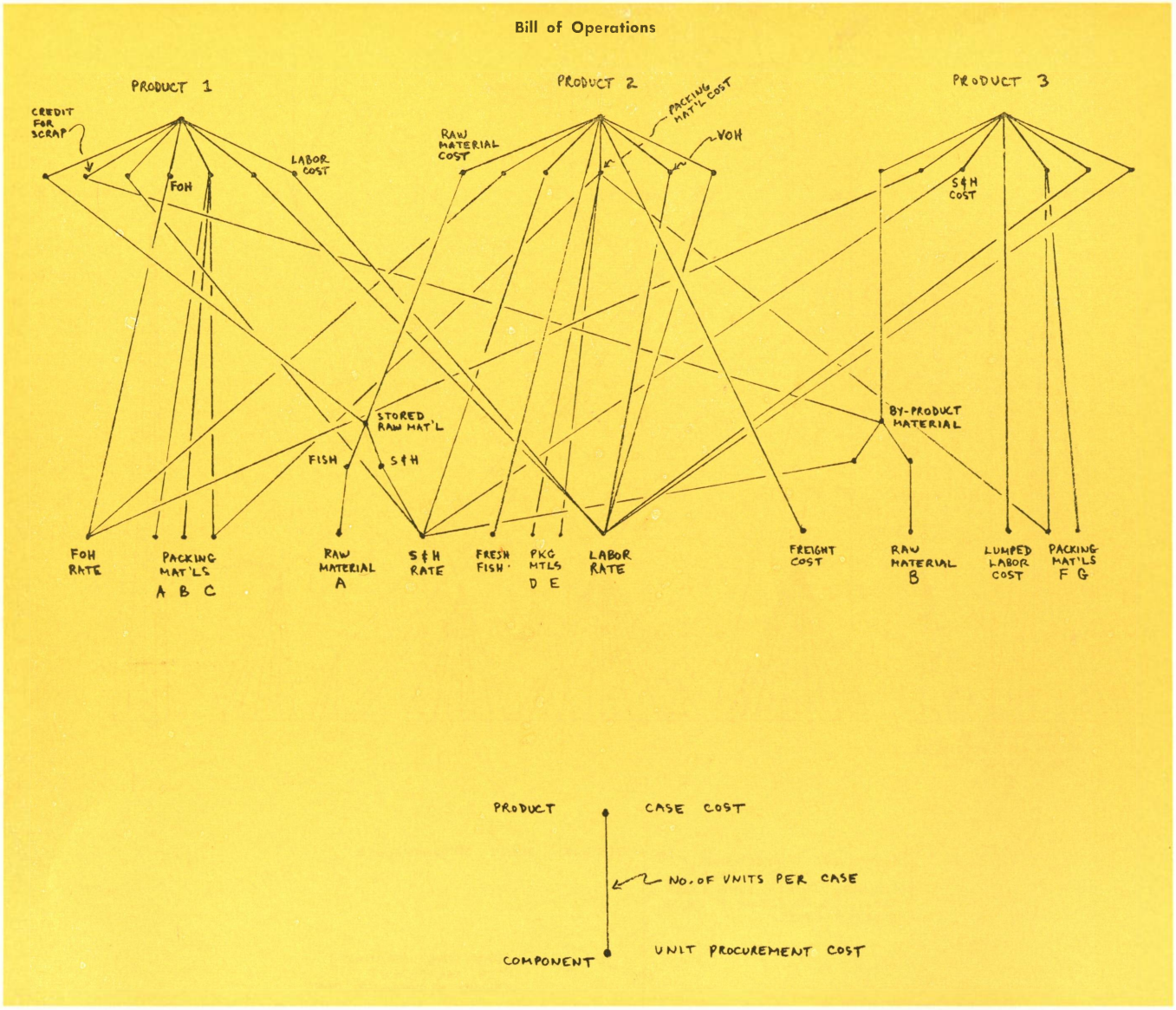


FIGURE 6



ply and demand. This production scheduling subsystem will then create the production plans that are fed into the main frame of the operations planning system in order to test their ability to meet demand for all products in all markets every week and their effect on total cost.

We expect many heartaches in developing this system — particularly in terms of storage capacity and processing speed problems, and even more in making it work for operating people. I am sure it will take five years and a larger computer to exploit the system fully. However, because of the versatility of network files, we can proceed gradually, we can introduce concepts step by step, and yet we can begin to benefit almost immediately from the system's tremendous value for many different aspects of management.

Property administration

Our third application of network files is not yet fully developed even in terms of our own thinking. This is our projected property administration system, illustrated in Figure 9 on page 53.

Our objectives in property administration are these:

(1) To record, maintain, and retrieve property lists by location and/or by custodial entity and/or by department charged (This inventory of properties is to include capitalized items, written-off items, and expensed items of sufficient interest.)

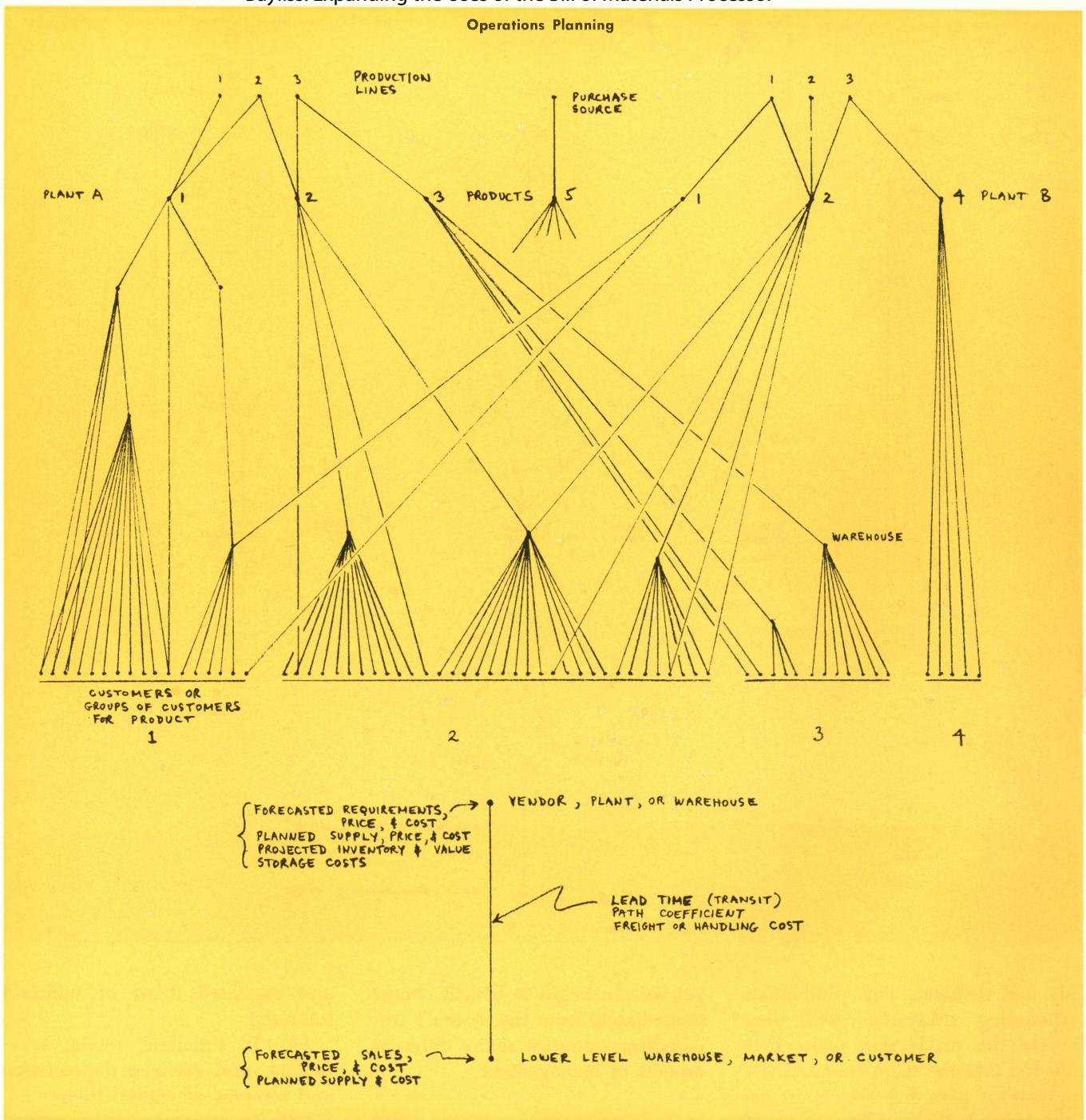
(2) To calculate, retain, accumulate, and retrieve depreciation and reserves of capital items

(3) To maintain, on a current basis, records of property *configurations*—such as properties that are assembled into a single production line or office equipment made up of modular units

(4) To produce, publish, and control equipment maintenance schedules and records of maintenance performed; also, to log equipment operating time or production figures for purposes of actual depreciation studies and maintenance control

(5) To control capital expenditures by means of matching au-

FIGURE 8



thorizations and estimates with actual purchases and with all actual costs pertaining to the purchase

(6) To create a job order system which gathers and controls all costs pertaining to individual construction orders or work orders, providing for partial capitalization without losing any significant element of job expense history (This generalized job order accounting aspect could form the basis of the entire cost accounting system in a different type of business. Note the similar function of our bill of operations.)

(7) To record and control leases and rentals, with a cumulation of payments and a continuous record of payout status

(8) To review and control insurance on all properties individually

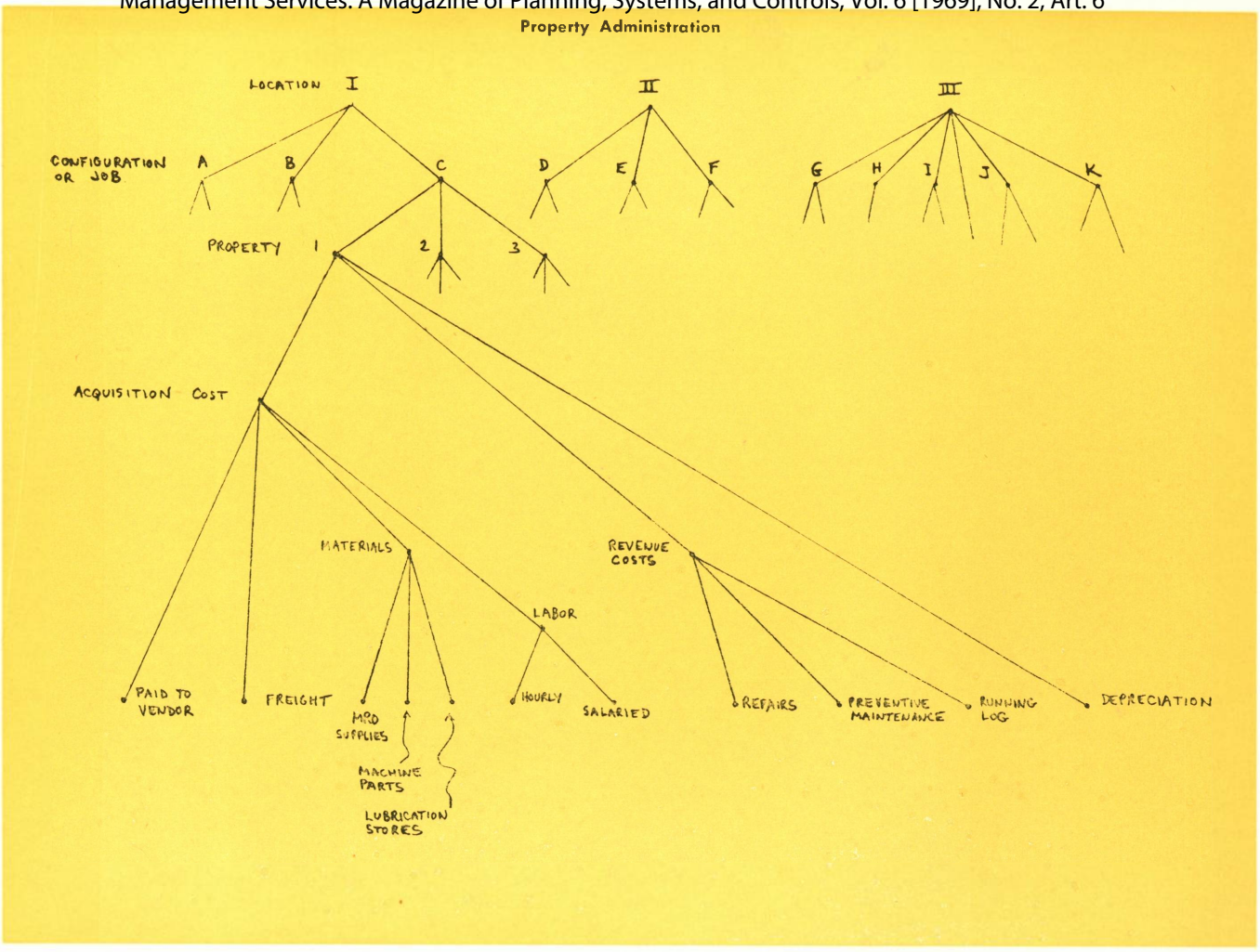
(9) To provide for control of off-premises properties (such as electric motors sent out for rebuilding)

(10) To provide for intracompany transfers of properties as well as trade-ins, sales of properties, etc.

Cost advantage

There is nothing unusual about these objectives. But we cannot afford major effort or major overhead in all or several of these areas; we must try to achieve them all with one computer subsystem and with an irreducible minimum increase in clerical staff. I think

FIGURE 9



that with network files we can kill all these birds with one stone.

Again, we shall start slowly and work in limited progressive steps. Our company has a tremendous need in all these areas, yet we cannot make this need a top priority computer application. Its purpose is more defense and control than direct profit making. The bill of operations, the operations planning system, and other programing projects will have far faster and greater payoffs. Yet by taking advantage of the powerful software available to us in the form of network file organization, we can move in this direction many man-years faster than we could with a conventional and piecemeal approach.

Conclusion

In all three of these applications the key is flexibility – ease of change of value and structure. File maintenance is more sensitive, more comprehensive, and administratively far cheaper than it would be with any other kind of file organization (that I know of) designed to serve the same purposes. It gives fingertip control, fine-tuning capability. And it gives the accuracy that can be achieved with nonredundant records serving multiple purposes.

Thesaurus approach

But its development is not easy. Although the software is written by the manufacturer, these packages take a great deal of study, and their programing structure is very difficult to master. Very good people are required to tackle something like this. No one should bother with the network file—I would say—if he has constant relationships between records or very few changes to make in existing records. In such a case, for instance, it might be better to use matrix algebra for explosions and retrievals.

However, when the nature of the operation justifies the effort, network files will provide a thesaurus instead of a dictionary. When someone knows the meaning of a word he wants but can't think of it, a dictionary does little good. Instead, he goes to Roget's *Thesaurus* to find a key word in the definition he is thinking of, or a related word, or even the opposite word, and the book will retrieve for him almost every conceivable cross-relationship. A thesaurus is an information retrieval system developed a hundred years before the computer. A network file approach may provide all the advantages of Roget's logic of association without hedging the incredible power of the computer.