Woman C.P.A.

Volume 47 | Issue 1 Article 6

1-1985

# "New Wave" of Decision Support Software: Is It For You? A Complement of the Information Center Concept

Bernard C. Reimann

# The "New Wave" of Decision Support Software: Is It For You?

## A Complement of the Information Center Concept

**By Bernard C. Reimann**

Do names like Express, IFPS, EIS, System W, Cuffs, Stratagem, or Xsim mean anything to you? If not, you may be missing out on an important, recent development in computerized decision support systems. If so, you may be among the growing number of executives who have had experiences similar to the following.

*A division controller in a large, decentralized financial institution has just completed next year's budget and is about to give the final copy to his division general manager, when he is interrupted by a phone call from corporate headquarters. There has been a last minute change in employee fringe benefits, which will have a significant impact on labor costs.*

*The year before a similar event might have come as a major shock and meant hours and hours of work to correct the many line items, consolidations, and allocations affected by the change. This time, however, it will be a matter of only an hour or so to change an entry in the new corporate model, rerun it, and print the revised departmental and consolidated divisional budgets. The controller will still be able to beat the deadline with time to spare.*

*A division general manager for a diversified manufacturer of capital goods is working on her five year strategic planning projections. While this is still not her favorite activity, she feels much better about it than she did a few years ago. She has a financial model of her division, driven by the unit sales volume of seven product lines. She starts by projecting these sales volumes, based on her understanding of industry and competitive trends, as well as other inputs, such as bottom-up forecasts from her sales force. Then the model turns the sales projections into five year forecasts of selected financial information including such items as profitability, net assets, and cash flow. Prespecified reports can also be generated quite readily.*

*Not long ago, it was a major effort to turn out just one five year plan and often there was considerable doubt about the validity of some of the resulting numbers. Now, thanks to the new computer model, she can easily develop alternative "scenario" plans, each based on a different set of assumptions. Typically she produces three alternative five year plans — pessimistic, optimistic, and normal. This makes the division manager less uneasy than she had been when she could develop only one set of numbers. Top management is happier because they get a more reliable set of planning projections, as well as a clearer idea of the degree of uncertainty implicit in the plans. What's more, headquarters also has a corporate model, including all of the twelve individual divisions plus corporate staff functions. The CEO, or the strategic planning staff, can use this model to play "what if" games to examine the impact of different strategies, and/or assumptions, on the bottom line. With careful research, plus some additional assumptions, it is also possible to analyze the financial impact of various combinations of acquisitions and divestitures.*

### Decision Support Systems

These two case studies illustrate some of the substantial benefits to be derived from a corporate decision support system (DSS). These DSS are not the same thing as MIS (management information systems). Where the major purpose of MIS is to give managers access to relevant information, the DSS is designed to select *and* manipulate information in ways which will help them make more informed decisions [Keen and Morton, 1978]. Such a DSS should permit the user to (1) Access relevant databases both internal and external to the firm, (2) Develop simulation models with which to manipulate and analyze large volumes of data, and (3) Display data and results in a variety of report or graphics formats.

### The Role of the Personal Computer

The newest generation of software for personal computers (PC's) offers some of these DSS features (e.g. Lotus 1.2.3, Multiplan). However, even the most advanced PC's fall considerably short on memory storage and computational power to provide a true, freestanding DSS.

A good PC's internal memory is more than adequate for relatively small databases with a limited number of variables and/or columns. However, it cannot handle the size and complexity of a large corporate database with multiple dimensions (e.g. line items and time periods classified by product, customer, territory, divisions, etc.) Even if a large amount of data could be downloaded to the PC, the process would be very time-consuming and risky from the standpoint of transmission accuracy.

Another problem with a DSS based on freestanding PC's is that of data security. With a mainframe system, experienced data processing professionals administer the process of managing and maintaining datafiles. With PC's this unfamiliar and tedious chore falls on the user who often does not do it very well.

Essentially the PC can manage a DSS, as long as it is fairly limited in scope, such as within a small firm or division. In a corporation with a large amount of data to be shared among many users, only a mainframe based DSS will be cost effective.

That is not to say that the PC cannot play an important role in a true corporate DSS. So far, however, its most effective role is as part of a network, rather than as an independent workstation. In such a DSS network, the PC can act as an interactive device for communicating with either the corporate mainframe, or with other mainframes, and/or PC's. It can also be used for limited local modeling, analyis, or display.

## "Fourth Generation" DSS Languages

A number of organizations have been using sophisticated DSS for quite a few years now. [Keen and Morton, 1978] Typically these systems were developed over a period of several years, at considerable expense, and with extensive involvement of the Data Processing function. The situations dramatized in the two case studies above are quite different in that the involvement of Data Processing personnel was limited to a relatively minor supporting role, and the time to develop the DSS was less than a year. This seeming miracle was made possible by the use of one of the new "fourth generation" financial modeling DSS languages.

These DSS software products are considered "fourth generation" languages because they are far easier to use than any of their predecessors (i.e. second or third generation languages such as COBOL, FORTRAN, or PASCAL). The developers of the new computer languages all had as their major goal to enable end users to build their own, custom-made DSS with little or no help from data processing. While not all seem to have been equally successful in this quest, several had managed to bring the enormous power of mainframe computing within easy reach of a rapidly increasing population of users. What's more, many of these new users are executives and professionals who either don't know, or don't want anything to do with, computer programming.

## Major Features of DSS Languages

Currently well over fifty vendors claim to provide mainframe DSS software (not including well over a hundred software products for PC's, such as Visicalc or Lotus 1.2.3, which are not in the same league as mainframe products with respect to power, flexibility, multidimensionality, etc.). So far only a dozen or so of these DSS languages appear to have had significant acceptance by industry, government, or educational institutions. All of these software products share a set of basic features (although to varying degrees) which make them particularly useful for do-it-yourself developers of DSS.

The Appendix gives a representative but by no means comprehensive selection of the DSS software products with which the author has some familiarity.

## Ease of Use

The bellweather of DSS languages is ease of use or "user friendliness" — especially for non-programmer users. This means that DSS builders can develop models, reports, or graphics with a relatively few, everyday English commands as opposed to the arcane hieroglyphics often needed by regular programming languages. Moreover, they don't have to concern themselves with the exact order, or "procedurality," of statements when building a model. Some languages even have extensive menus or prompting sequences to guide inexperienced users. On-line computer assisted tutorials are also available in a few cases. All vendors offer some sort of basic training for new users.

## Modeling and Interrogation

The real "guts" of a DSS language is its ability to model a variety of decision situations. All of the software products in the Appendix allow the user to create a set of complex equations to describe a given situation. However, the different languages vary greatly in their power and flexibility for representing complex situations. Some are limited in the number of dimensions they can handle beyond the basic two spreadsheet dimensions of time and line items or accounts. There are differences also in the ability to handle time as a special dimension, or in handling such added "complexities" as probabilistic or stochastic variables.

Perhaps the most dramatic and useful feature of these DSS languages is their highly flexible model interrogation capability. This feature allows the user to get virtually instantaneous answers to any number of "What If" or sensitivity interrogations. Users can substitute various different assumptions for one or more input variables (e.g. sales volume, price, or costs) and see the impact on the bottom line. Another useful feature is "goal seeking." Here users can specify a given end result value (e.g. profitability) and determine the input value(s) required to be able to attain the desired end result. This, of course, would be a particularly valuable feature when an executive is trying to come up with some idea of the alternatives available to achieve a particular objective. Similarly, the "what if" capability allows the user to test the effects of different events (such as new governmental legislation, changes in consumer sentiments, etc.) on the performance of the organization.

## Forecasting and Statistics

All of the better languages have some built in basic statistical and time series analysis functions. Only a few are able to handle sophisticated econometric forecasting models, including such functions as ridge regression and SABL seasonal adjustment. Generally speaking, those languages which excel in econometric forecasting are somewhat weaker in financial modeling.

## Reporting and Graphics

Many of the new languages really shine when it comes to effective display of information. Typically they make it very easy to develop customized reports or graphs. Not unexpectedly it is in this graphics capability where some of the key differences between competing products can most easily be seen. A few offer almost an unlimited number of options with respect to such things as individual labels for each of several different curves or bars, "exploding" pie charts, or highly customized legends. Others

are much more restrictive, allowing only a limited number of standard plot options. (An example of a convenient, easy to use, but highly limited graphics package is the one provided in Lotus 1.2.3.)

## Communications

A strong and significant recent trend is that of decentralized, or distributed computing, using a combination of a main frame computer and a large number of PC's throughout the organization. A critical consideration in this kind of system is the degree to which the DSS software facilitates or hinders communications between mainframes (both internal and external to the organization) and PC's. Several of the DSS languages listed in the appendix have developed effective and relatively easy to use linkages between mainframes and PC's.

## Data Management

Most of the DSS languages in the Appendix do not offer any built-in data management capabilities. This generally does not present a major problem until the DSS applications get fairly large and complex, with many different users. Then problems such as data file proliferation, data security, and ready access to "foreign" databases (either internal or external) start to surface. This is when users begin to appreciate the value of a data management feature. For most of the languages in the Appendix, users either have to manage their own datafiles or enlist the help of data processing and one of the commercially available data management software products, such as Info, Focus, or Adabase. Unfortunately, most data base management systems are not particularly user friendly. But even if they were, the process of linking the DSS' language with one of these data managers is very complicated and not at all "friendly," so that the user would require considerable help from data processing.

## Command Language

These fourth generation languages typically have another important feature which lets more advanced users expand and modify the language to suit their changing requirements. With this command language facility, the experienced develop a DSS for particular applications which can then be used by relatively inexperienced people. This can be done by building a "command file" which includes a set of predetermined prompts and/or menus to guide the user. For example, a staff member familiar with the DSS language can easily design a customized DSS for his boss (say the CEO) to make certain predetermined inquiries such as "How would a 10 percent drop in sales of hard core widgets for the Eastern division affect our corporate net earnings next year?" Or "What would our labor costs have to be in our Cleveland plant to give us a corporate return on sales of 5 percent?"

## Additional Considerations

The above seven features have been fundamental to the promise of these new DSS software products to "finally let management get its hands on the data." [Wolf and Treleaven, 1983]. It is these features which have made it possible for practicing line and staff executives with little or no prior familiarity with computers to tap into the virtually unlimited power of mainframe computers. In addition, there are three other factors to consider when comparing or evaluating the various competitive products in this area — cost, vendor support and hardware considerations.

## Cost Considerations

While the potential benefits of these fourth generation DSS languages appear undeniable, they naturally do come at a price, and a rather substantial one too. Two important cost components must be considered: (1) the initial cost of obtaining the use of the software and (2) the cost of installing the software and then getting it up and running on the in-house computer system.

*Initial Cost.* Most of the fourth generation languages in the Appendix are available for use either on a timesharing basis or on an in-house licensing arrangement. Timesharing has advantages and disadvantages similar to leasing equipment or hiring temporary personnel. Often it is a good way to get started with minimum risk, since most vendors have arrangements where a new customer can try out software on a timesharing basis for a period of a few months.

The current trend is clearly toward in-house computing. This arrangement typically includes an initial licensing fee to purchase a computer type containing the source code for the language. Then there is an annual maintenance fee of about 12-15% of the initial license fee to cover periodic "updates" or product improvements, as well as training, consulting, and other maintenance services.

Licensing fees vary a great deal, especially now that competition appears to be heating up. Some vendors offer a "complete" system or package for anywhere from $50,000 to $300,000. Like the automakers, many have "unbundled" their prices and offer a basic "no frills" system (say modeling only) for as little as $5-10,000, but if users want other options (e.g. forecasting, reporting, graphics, or communication) they still may wind up paying $100,000 or more.

*Installation and Running Cost.* A major cost for any new system is the time and effort required to get it to the point where it can be used for its intended purpose. For DSS software this cost typically outweighs the initial license fee by a substantial amount. Therefore, it pays to make sure the "right" software is installed in the first place [Reimann and Waren, 1984]. If the language has the key features needed by the prospective users, and is "truly user friendly" then the path toward effective use should be relatively free of major obstacles. A strong, supportive vendor organization is also a big help in implementing an in-house DSS. (Hardware compatibility issues are also important, but are not considered here due to the highly technical and complex nature of this issue).

## Vendor Support

Another factor which should not be overlooked when considering the acquisition of a DSS language is the quality of the vendor support organization. Some characteristics to look for include: quality and availability of documentation, training, consulting and frequency of major product enhancements, or "updates."

Since these software products are extremely complex and constantly being enhanced, it is also important to have a vendor organization which is able to support the product in all of these areas and will continue to be able to do so over the long term. Many of the firms offering DSS software have been in existence for only a few years

## APPENDIX

### Selected DSS Software Products and Vendors

| Product | Description | Vendor |
|---|---|---|
| CUFFS 88 | A non-procedural/procedural financial modeling language for IBM and DEC mainframes. | Cuffs Planning & Modeling Ltd.<br>201 East 87th Street, Suite 26C<br>New York, New York 10028<br>(212) 534-6404 |
| EIS | A comprehensive, integrated software package available through time-sharing or in-house on IBM mainframes. | Boeing Computer Services, Inc.<br>Division of The Boeing Company<br>P.O. Box 3707<br>Seattle, Washington 98184<br>(206) 251-3190 |
| EMPIRE | Comprehensive financial modeling package for DEC or IBM mainframes. | Applied Data Research, Inc.<br>Route 206 and Orchard Road<br>Princeton, NJ 08540<br>(201) 874-9000 |
| EXPRESS | A powerful, multidimensional modeling package for IBM or Prime computers. Time-sharing is available as well as a link to LOTUS 1.2.3. on PC's. | Management Decision Systems, Inc.<br>20 Fifth Avenue<br>Waltham, MA 02254<br>(617) 890-1100 |
| FAME | A procedural language for forecasting and financial modeling on DEC and IBM mainframes. | Gem Net Software Corporation<br>2175 W. Stadium Blvd.<br>Ann Arbor, Michigan 48103<br>(313) 663-4333 |
| FCS-EPS | Financial modeling software package available via timesharing or for in-house use on a wide variety of hardware. | EPS, Inc.<br>One Industrial Drive<br>Windham, NH 03087<br>(603) 898-1800 |
| IFPS | Non-procedural, English-like modeling language for use on wide variety of mainframes. Micro verson available also. | Execucom Systems Corp.<br>3410 Far West Blvd.<br>Austin, Texas 78731<br>(512) 346-4980 |
| MODEL | Comprehensive, modular, and procedural financial modeling language. Available for a variety of mainframe, mini, or micro computers. | Lloyd Bush & Associates<br>14679 Midway Road<br>Dallas, TX 57234<br>(214) 233-4549 |
| STRATAGEM | Versatile, multidimensional modeling package available for DEC and IBM mainframes. | Integrated Planning, Inc.<br>338 Newbury Street<br>Boston, MA 02115<br>(617) 267-5914 |
| SYSTEM W | Powerful multidimensional financial modeling language available through time-sharing or for in-house IBM mainframes. "Micro W" version for PC with linkage to mainframe. | Comshare, Inc.<br>P.O. Box 1588<br>Ann Arbor, Michigan 48106<br>(313) 994-4800 |
| XSIM | Comprehensive forecasting and financial modeling language for IBM mainframes. Can access vendor's own databases, and link to PC's. | Chase Econometrics/IDC<br>486 Totten Pond Road<br>Waltham, MA 02154<br>(617) 890-1234 |

and are quite small and financially weak (several have fewer than 20 full-time employees, for example). On the other hand, at least two of these firms have been acquired by large corporations recently. This would at least give them the financial backing to stay in business over the long term.

The various vendor organizations also differ considerably in their mix of support for training new users, and providing on-going client consulting services. These services can make all the difference between a smooth and relatively trouble-free software installation and an expensive exercise in frustration. References from other firms using the DSS software can be particulary useful in getting some idea of the vendor's level of support here. A visit to the vendor's headquarters prior to purchase is also suggested. If the vendor has a local support office, so much the better. This local office and its personnel should be checked out especially carefully, since this is where the bulk of the ongoing support will come from.

## Picking the "best" DSS Language

Finding the best DSS software product for a given user can be an exceedingly difficult and time-consuming process. The applications for which these products were designed are extremely varied and complex, and the software products naturally reflect this complexity. Due to the newness and rapid change of this field, there is little guidance available in the literature with the exception of a few articles which have appeared very recently [Bergstrom, 1982; Wolf and Treleaven, 1983; Meador and Mazger, 1984; Reimann and Waren, 1984]. The latter two articles provide particularly extensive lists of the key features of different DSS languages, as well as guidelines for their comparative evaluation.

Essentially this evaluation process boils down to the following:

1. Determine which features the prospective users (a) absolutely have to have, and (b) would just like to have.
2. Use this list of features to screen out those products which do not have one or more of the "must have" features.
3. Rank the survivors on the degree to which they incorporate both essential and "nice" features.
4. Arrange vendor visits and comparative "benchmark" tests for the four to six top ranked products.
5. Pick the one or two "best" of the remaining products and bring them in house for a "pilot test" of three to six months duration. The survivor of this last trial should stand an excellent chance of truly being the "best" product for the prospective users' application(s).

One final bit of advice on this selection process. Since this is a highly complex and specialized area for which few corporations have the expertise, the use of a reputable, experienced consultant who specializes in DSS software applications is strong-

ly recommended. The cost will be minimal when compared to that of making the wrong choice.

However, if you prefer to make your own evaluation, here are a few suggestions which might make the task easier and more likely to yield satisfactory results. If you know of any organizations similar to yours which have recently purchased one of these DSS software products, the people involved in its selection may give you some valuable guidance. The vendors themselves can provide useful (though obviously biased) information on how their products stack up against the competition. Your local university is likely to have some experience with one or more of these software products, and may be able to provide some valuable (and relatively impartial) information.

A particularly useful device for comparing the performance of a limited number of different DSS products is a set of "benchmark" tests. An example of such a benchmark is some part of a potential application for the DSS, such as a financial model for a division or department. The test should involve as many of the critical DSS features as possible (e.g. data transfer, sensitivity testing, statistical analysis, report writing, graphics). Each vendor would be asked to complete the same benchmark test and provide documentation of all steps required. If possible, some rundown of resource usage should also be requested. This may not be as meaningful, since resource usage is a function of the particular hardware and operating system configuration used, and this typically differs considerably from vendors to vendor.

Finally, a consulting firm, Real Decisions, Inc., has put together a very useful handbook, *Financial Modeling Decisions,* which gives a summary of the capabilities of twenty or more DSS software products, including their performance on a set of benchmarks. This handbook, or information about specific products and vendors, is available for a fee.

## The "Information Center"

The DSS software products described in this article have a great deal to offer to executives in a variety of functions requiring the management of large quantities of information. Organizations which acquired these products for in-house use have typically experienced significant improvements in timely access to information in a form that management needs and can use.

This new type of software is rapidly becoming an effective complement of the "Information Center" concept being promulgated by IBM. Essentially this concept involves a transformation of the traditional data processing function from active involvement to merely providing technical and hardware support to users of information. The objective is to put total control of computing and information processing in the hands of the end-user. If your managers have problems getting their hands on the data they need, when they need it, perhaps it's time to look into what this "new wave" of end user computing software can do. Ω

### REFERENCES


Bergstrom, L.P. "Selecting Financial Modeling Software," *ICP Interface,* A&A (Autumn, 1982), pp. 20-26.

Keen, P. and Scott Morton, M.S., *Decision Support Systems: An Organizational Perspective,* Addison Wesley, 1978.

Meador, C.L. and Mezger, R.A., "Selecting an End-User Programming Language for DSS Development," *Proceedings of DSS-84,* New York, N.Y., May, 1984.

Reimann, B.C. and Waren, A.D., "A Systematic Approach to Selecting the 'Right' DSS Software," unpublished working paper, Cleveland State University, Cleveland, Ohio, March, 1984.

Wolf, C., and Treleaven, J., "Decision Support Systems: Management finally gets its hands on the data," *Management Technology* (November, 1983), pp. 44-50.



**Bernard C. Reimann, DBA,** *is professor of management at Cleveland State University. He earned his B.Ch.E. from Cornell University and his MBA and DBA from Kent State University. He has been on the faculties of Drexel University and the Wharton School of the University of Pennsylvania. He has written numerous articles for professional journals.*