DOCUMENT RESUME

ED 052 788 LI 002 932

AUTHOR	Barnett, Arnold
TITLE	[Selected Papers on Effective Data Systems Development.]
INSTITUTION	Barnett Data Systems, Rockville, Md.
PUB DATE	[71]
NOTE	60p.; (0 references)
AVAILABLE FROM	Barnett Data Systems, Suite 507, 1010 Rockville Pike, Rockville, Maryland 20852 (No Charge)
EDRS PRICE	EDRS Price MF-$0.65 HC-$3.29
DESCRIPTORS	Automation, *Computers, *Data Processing, *Information Processing, Information Systems, Management, *Management Information Systems, Systems Analysis, *Systems Approach

ABSTRACT

The four papers included in this document are: (1) "Preparing Management for MIS" presented at the GUIDE-32 Meeting, May, 1971; (2) "Training Management for MIS" presented on March 17, 1970, to the Central New York Chapter of the Association for Systems Management; (3) "How to Unlock the Computer's Profit Potential" presented at the 1969 Spring Seminar of the Steel Valley Chapter of the Association for Systems Management; and (4) "The Process of Effective Data Systems Development" presented at the 1968 Data Processing Management Association Convention. (MM)

[Selected Papers On Effective Data Systems Development]

By

Arnold Barnett

Barnett Data Systems
Suite 507
1010 Rockville Pike
Rockville, Maryland 20852

PREPARING MANAGEMENT FOR MIS

Presented at the GUIDE-32 Meeting, May, 1971

by

Arnold Barnett

Barnett Data Systems
Suite 507, 1010 Rockville Pike
Rockville, Maryland 20852
Telephone: (301) 762-1288

"Preparing management for MIS" could, to the uninitiated, conger up a vision of an executive being made comfortable in his easy chair and his data systems manager wheeling a cathode ray display device in front of him. This vision implies that the executive is sitting back and relaxing during the development of his MIS and only before the system goes operational will he be taught to use his new system. Note that I used the word "uninitiated." You in GUIDE and I in my business have been through too many systems efforts to know that developing a successful MIS cannot be done without the meaningful involvement of the managers who will use the MIS. Many cancelled efforts and poorly designed MIS attest to this fact.

Therefore, to develop an MIS which will go operational on time, do what it is supposed to, and be accepted by those for whom it was designed requires that both top and middle management understand, accept, and play their roles in their system's development.

The concept of management involvement is all well and good and there is lots of discussion about it, but what has to be done to make it a reality? This talk, which is based on my own experience and knowledge, will first answer the question of what must be done to get management involved in the development of their own MIS and then go on to explain how to best use this involvement to produce a good MIS.

The first consideration is that management is going to have to spend some time, preferably three days, to learn: a logical systems development process that allows for their participation in systems efforts; the interacting roles they and the systems people play during these efforts; some non-detail techniques to play their roles; and, most important, be so sold on the program that they will put into practice what they have learned.

Before going on I want to stress that this training is primarily systems training, management oriented, and delivered in English. Technology is secondary. I

believe one of the biggest deterrents to meaningful management involvement is education for non-systems people that focuses primarily on binary arithmetic, computer concepts, software, what the computer will do for you, hands-on, etc. It either "turns them off" because they equate involvement with detail, or makes them into "ADP experts" and, as such, get too involved in the wrong things at the wrong time.

Back to the training program. We see that the first consideration is having a logical systems development process that allows for management's meaningful involvement in their own systems efforts. On the opposite page is illustrated such a process, called EFFECTIVE DATA SYSTEMS DEVELOPMENT.

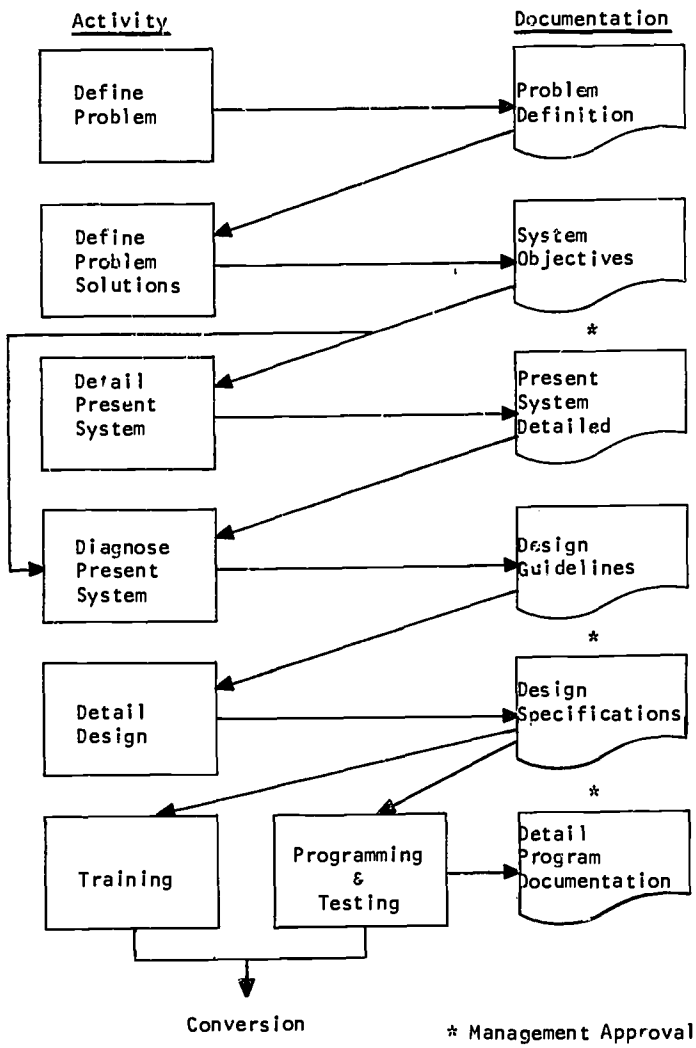
The first activity calls for discovering and documenting for the particular system under development its current and future problems and needs. Next, the solutions to these problems and needs are formulated and documented generally as to how they will be solved, to what degree, and the time, money, and people needed to do the job. These system objectives are approved by both systems and operating management before being taken to top management for their approval.

The current system is then described and documented in detail. Guided by the system objectives, the current system is diagnosed and reasonable changes that can be made to the current system are documented in the design guidelines. The design guidelines, which detail and redefine the system objectives, are approved by systems and operating management and then taken to top management for their approval.

The detail design of the system is then accomplished and the resulting documentation is called the design specifications. As with the system objectives and design guidelines, the design specifications are approved first by systems and operating management and then by top management.

Approval of the design specifications triggers the programming, testing, and debugging effort and also the training of the

The Process of EFFECTIVE DATA SYSTEMS DEVELOPMENT



operating personnel. When these are completed the system is converted.

At this point I am sure you are intellectually agreeing that this is a good process for systems development. Also, I am sure you will agree that most systems efforts are not done this way. I would like to submit that most efforts begin with some firm idea of the solutions before all of the problems are known and where little analysis and design work are done before programming. It is during the programming that the lack of good problem definition catches up with the effort in the form of changes and additional requirements. Consequently, programming becomes a process of constant analysis, design, programming, change, reanalysis, redesign, reprogramming, etc.

Finally a system goes operational and does not do what it is expected to; heavy maintenance and redesign work is done and eventually the system settles down to something acceptable to the operating people. It is amazing, but this is typical of the history of what most people call their "successful" systems and, sad to say, this is what many operating and systems people think must be endured to have any automated system. Well, it does not have to be this way, and there are success stories to back up this statement.

We have already looked at a logical process of systems development, now let's consider the interacting roles that operating and systems people play throughout that entire process.

First, for every systems effort a formal or informal systems team should be established, composed of systems and operating people. Individually these team members represent their own departments and collectively they are responsible for the systems effort. The team leader will also represent top management. Note that the team is primarily responsible for the effort, not the systems or operating departments.

Second, the work carried on by the operating people should

be meaningful, which means that they should not be expected to know and do the kind of detail work performed by systems analysts and programmers. Throwing them into that kind of work, for which they are ill prepared, will see them either drift off the team or, if they stay, become estranged from their home departments. The reason for the latter is that the operating people back in their department will look upon them as "traitors" who have become just like the people in the Data Processing Department.

One of the most meaningful jobs for the operating team members is to maintain an open line of communication between the team and their bosses and co-middle managers back in their departments. Through this medium, operating management will be constantly updated as to progress and/or problems of the team. Also, management will be meaningfully involved when their representative encourages them to make relative suggestions and decisions. Note that this communication is constant throughout the effort. Therefore, these relative suggestions and decisions will be considered as design inputs not as cause for delay, redesign, and reprogramming. The open line of communication also applies to the systems members of the team who are keeping their bosses, the programming manager, and the computer operations manager apprized of what is being considered and allowing them, as well, to offer suggestions and make decisions.

In addition to being communicators, the operating members of the team will help in defining the problem, establishing the system objectives, design guidelines and design specifications. They will be of material help in some of the areas peripheral to the automated portion of the MIS, such as: manual system changes, policy changes, reorganization, validating existing files, training, developing better supervision and operating controls, etc. Also, because of their presence, the man and the machine portions of the system will be designed concurrently. This concurrence is important. Today most poorly functioning MIS were designed first as automated systems and then consideration was given to the people part of the MIS.

In addition to their job as communicators and non-detail computer system designers, the operating members of the team will be primarily responsible for the training and briefing of the operating personnel. This training will be done prior to conversion during the time when the computer programs are being written and tested. Having the operating members of the team train their own people goes a long way toward reducing resistance to change in the operating department.

Up to this point we have highlighted the process of systems development and the interacting roles that operating and systems personnel play in that process. Let's get into a little more detail and discuss just how a systems effort should be started. This is very important because many of our current-day failures can be traced back to a poorly conceived beginning.

In this connection, I want to discuss three very common concepts that I believe are injurious to good MIS development. These are: initiating the systems effort by describing the present system in detail; initiating the systems effort by first researching the company's objectives, policies, long-range plans, etc.; and, asking management to state their information requirements. Note that I said that these three practices were injurious to good MIS development.

When an effort starts with a detail description of the present system it gets into too much detail too soon; the persons doing the work are soon unable to "see the forest for the trees" and, consequently, the information requirements of the system are not determined at the beginning but are discovered later in the effort when their inclusion into the system are cause for redesign, reprogramming, delay, etc.

Appreciate that I am saying not to begin the systems effort with a detail description of the present system but to do this description <u>after</u> the problems and solutions have been

generally defined. The added benefit of describing the present system after the system objectives have been approved is the analysts will detail only where it is felt that there will be system change. This means a much more directed effort based on good judgments as to what can be done to solve problems not just a fishing expedition.

The second concept which I believe to be injurious to good MIS development is initiating an effort by first nailing down and documenting the organization's objectives, goals, policies, long-range plans, etc. Again, I am not saying that this should not be done, I am saying not to do it as the very first step in the systems effort. It should be done after the problem definition, during the time when the solutions to the problems are being determined. Much time will be saved because it will be done in a specific context and its being done will insure that the problem solutions are travelling in the same direction as the organization. Also, with management involved, getting these often nebulous areas defined will be relatively easy.

The third and most injurious concept is the one that says that management should define their own information requirements. Theoretically it sounds good; however, in practice it does not work. The reason it does not work is that management is required to define their own current and future information problems and needs relevant to the MIS and also to define their solutions to these problems, i.e., the requirement for a monthly report on so and so. I submit that management should not be expected to be so sophisticated that they can systematically define their own problems and needs, and they are even less able to translate these information problems into system requirements.

The proof of what I am saying is in the countless number of requirement letters, signed in blood by operating management, that are materially changed prior to the resulting system conversions. Some of those subsequent changes were wrought by the operating managers themselves and others by the systems department because the stated requirements were "unrealistic." In many cases any similarity between the original

requirements and the resulting system are purely coincidental.

Therefore, I suggest that systems people interact with management to help management define their problems. The burden is on the operating manager to "cry his heart out" during the problem definition with the systems person keeping their interaction in context and out of detail. After the problems have been defined the burden will shift to the systems team to come up with reasonable solutions. Remember that not all of the solutions burden is on the systems members of the team because not all of the solutions to information problems are in automation. Therefore, both systems and operating people should participate in the problem definition and in formulating the system objectives. In addition, this meaningful participation will continue throughout the entire effort.

Let's recap:

- MIS development cannot be done without management's meaningful involvement.
- Management should take the time to understand the process of systems development; their roles in this process; learn techniques to play their roles; and, most important, be so sold on the program that they willingly practice what they have learned.
- Follow the process of EFFECTIVE DATA SYSTEMS DEVELOPMENT with its six activities and corresponding documentation and three management approval points.
- Establish a systems team composed of operating and systems personnel.
- Be careful that work assignments given to operating personnel be relevant to their knowledge and capabilities, i.e., no detail systems work.
- All members of the team should keep the lines of communication open to their respective departments and, when appropriate, involve both operating and systems departments' management by having them offer suggestions and make decisions.

- Design the computer portion and the rest of the system concurrently.

- Operating people should train their own personnel, one benefit of which will be to materially reduce resistance to change.

- Do not begin a systems effort with a detail description of the present system; do this after the system objectives have been approved.

- Do not begin a systems effort by documenting the organization's objectives, policies, long-range plans, etc. Do this after the problem definition and in conjunction with the formulation of solutions.

- Do not begin MIS efforts by asking operating management for their requirements. Efforts should begin with a problem definition prepared primarily by operating management aided by members of the systems team. Subsequent to defining the problems the solutions will be formulated primarily by the systems team with advise and consent from operating management.

Other highlights of this process are:

1. There is "open season on changes" from the time the system objectives are being formulated to just prior to the approval of the design specifications; however, once the design specifications have been approved, "hunting season is over."

2. Impasse situations that may occur between operating departments and/or the systems departments will be recognized and solved prior to the final sign-off of the design specifications. Appreciate that this process precludes many of the impasse-type situations faced today; however, if they do occur, they are argued and solved before programming, not during programming.

3. Under EFFECTIVE DATA SYSTEMS DEVELOPMENT, the design specifications will contain much of what is today

found in program documentation. This means that all of the detail analysis and design work is completed prior to programming. Systems management should not sign-off on the design specifications unless they feel that they contain all of the information necessary to allow their programmers to "program in peace."

4. 51 to 75% of the time in a systems development effort will be spent from the initiation of the effort through the approval of the design specifications. The remaining 49 to 25% of the time will be in programming, testing, training and conversion.

Let us review the major benefits that are being gained today by organizations that are: properly employing operational people on systems teams, following a step-by-step process of systems development and beginning their efforts with good problem definitions:

1. Their resulting systems are more responsive to the users because the persons for whom the systems are being developed are involved in such activities as defining their own problems, describing their own present system, designing the new system and training their own personnel prior to the conversion. Many of the problems that might arise during the programming, conversion and system operation are precluded by this prior involvement.

2. Subsequent redesign and maintenance to their systems are minimized due to operating department involvement in the design of the new system.

3. Systems development efforts are less hectic and there are "no surprises" for the operational departments when they review the interim and final designs of the new system. This is because the users themselves have been participating in and guiding the effort from its inception. Also, "no surprises" helps insure on-time conversions.

4. The conversion and subsequent operation of a new system is smoother because the operating personnel who have played a part in its development consider it their

own system. This has a favorable impact on the personnel of the operating department. They consider the new system as one designed not by the systems department but by their own management, a big factor in reducing their resistance to change prior to, during, and after the system is converted.

5. Operating management takes a proprietary interest in their new system and does not abrogate their responsibility toward its automated portion. They do not consider the automated portion as belonging to the Data Processing Department. Also, there is good cooperation between the user departments and the computer center during the operation of the system. All departments tend to consider themselves as belonging to a partnership rather than looking at each other as antagonists.

In closing, management can be prepared for MIS, many have been prepared, and there are organizations that are today reaping the benefits of on-time, workable, user-accepted systems.

* * * * *

The following may also be of interest:

Speech Reprints (Free)

"The Process of Effective Data Systems Development," presented at the 1968 Data Processing Management Association Convention.

"How to Unlock the Computer's Profit Potential," presented at the 1969 Spring Seminar of the Steel Valley Chapter of the Association for Systems Management.

(over)

"Training Management for MIS," presented in March 1970 to the Central New York Chapter of the Association for Systems Management.

Additional copies of this speech.

Textbook ($15.00)

EFFECTIVE DATA SYSTEMS DEVELOPMENT

Seminar (3-day, in-house and public sessions)

EFFECTIVE DATA SYSTEMS DEVELOPMENT

To obtain additional copies of this speech, the other speeches or textbook noted above, or information on the seminar, write or call:

Barnett Data Systems
Suite 507 - 1010 Rockville Pike
Rockville, Maryland 20852
Telephone: (301) 762-1288

TRAINING MANAGEMENT FOR MIS

Presented on March 17, 1970, to the
Central New York Chapter of the
Association for Systems Management

by

Arnold Barnett

Barnett Data Systems
Suite 507 - 1010 Rockville Pike
Rockville, Maryland 20852
Telephone: (301) 762-1288

About two years ago a county government experienced a very painful implementation of an automated tax billing system. Bills were rendered months late and, when they finally got out, a substantial number were wrong and misaddressed. When the county auditor was asked by a newspaper reporter why all this had happened, the auditor's reply was "When you automate it is always hell for the first three years."

I like to tell this story because it highlights two problems that continue to face us in the development and operation of data systems. One is that most development efforts result in systems: that are not well received by the users, miss their operational target dates, are difficult to operate, require an inordinate amount of redesign immediately after they go operational, etc.

The second problem is that both user management and their data processing specialists have gone down this road so many times that, like the county auditor, they think this has to be endured to have an automated system.

Well, it just does not have to be this way. Decent, on-time data systems can be developed if we just go about it in the right way. Moreover, if we do not go about it in the right way, when we develop our management information systems (MIS), we will be in for more trouble than we have had with our other systems efforts. The administrative, financial and operating systems we have been working on have a manual or automated precedent system to guide us; however, with an MIS, there is very little in the way of a precedent system. The current systems development processes we employ to improve our administrative, financial and functional systems are just not adequate to tackle the job of developing an MIS. And, as I said before, these current systems development processes are not really adequate to properly do the jobs we have been doing.

One thing that has done more to hamper good systems development work is the absence of user involvement in the initiation and development of their own data systems. In some cases we can blame management for not responding to this request by their systems people and, in other cases we can blame the systems people for not requesting this involvement; however, pointing the finger at each other is not going to get us anywhere at this time.

As noted before, we can muddle along without management's involvement when we are dealing with precedent-type systems; however, this same process for the systems development of an MIS will produce a disproportionate amount of fiascos - systems under development that are so bad they never go operational - and operating failures - systems that go operational and are maintained only to give a negative or, at best, a very little return for the time, money and people invested in their development and operation.

I believe the answer for our current, as well as future, systems development efforts is a process of systems development believed in and supported by top management that requires the involvement of the persons for whom the systems development effort is being conducted.

The crucial point is how do we get top and middle management to believe in such a program (some systems people, too) so that they will take some of their valuable time to participate in the development of their own data systems.

The answer is that this particular process for systems development must be sold via an effective training program. It is one thing to train management in the basics of data processing such as binary arithmetic, programming, computer concepts, hardware and expose them to the computer via "hands-on" training and demonstrations of what the computer (and its remote display device) can do. However, it is another thing to spend three days with management convincing them to meaningfully involve themselves in the development of their own data systems. Convincing them by explaining:

(1) The process of systems development from initial planning through analysis, design, implementation and subsequent evaluation of the operating system;

(2) The interacting roles that both management and systems personnel play throughout the entire effort; and,

(3) Then offer some systems techriques that management can use without taking any additional training.

This latter type of training focuses on what management must do to help in the development of their own systems and makes them comfortable and willing to do so; whereas, the former training stresses mainly what systems people do and, in some cases, impedes management's meaningful involvement.

To review:

(1) We have had and continue to have troubles developing our precedent-type data systems.

(2) If we develop our MIS in the same way we are doing our other systems development work we will be in for even more trouble.

(3) The majority of the problems in systems development stem from the fact that managers, top and middle, do not meaningfully involve themselves in the development of their own data systems.

(4) To develop good management information systems, top and middle management will have to be convinced to involve themselves in the development of their systems and be trained to play their respective roles in the systems development effort.

(5) This is best done by a seminar that focuses on systems development not hardware, software, hands-on training, and demonstrations of what the computer can do for you.

Let's be more specific about the advantages of management's involvement in the development of their own data systems:

(1) With management involvement, the resulting system will be more responsive to the users because the persons for whom the system is being developed will be involved in such

activities as defining requirements, describing their own present system, designing the new system and training functional personnel prior to the conversion. Many of the problems that currently arise during the programming, conversion and system operation are precluded by management's prior involvement.

(2) Subsequent redesign and maintenance to the system will be minimized due to user involvement in the design of the new system.

(3) Top management will be more likely to approve and support a "joint" systems effort rather than a systems effort proposed by either the systems department or the using department.

(4) The systems development effort will be less hectic with much less feedback because there will be "no surprises" for the user departments when they review the interim and final designs of the new system. This is because the users themselves will have been participating in and guiding the effort from its inception. "No surprises" will also insure an on-time conversion to the new system.

(5) The conversion and subsequent operation of the new system will be smoother because the users who have played a part in its development will consider it their own system. This will have a favorable impact on the personnel of the users function. They will consider the new system as one designed not by the systems department but by their own management, a big factor in reducing their resistance to change prior to, during and after the system is converted.

(6) Management will take a proprietary interest in the new system and not abrogate their responsibilities toward the automated portion of their data system. This abrogation is noted when functional management considers "their" automated systems as belonging to the Data Processing Department. In such a case, the payroll manager would tell the data processing manager

that it is his problem (the DP manager's), that the time and attendance information is inaccurate and that he should write the operating departments asking for improvements in their time and attendance reporting. This is not the "mothering, protecting and defending" role the payroll manager should be playing. Such abrogation of management's responsibilities leads some to conclude that the man who controls the MIS will control the organization.

(7) Last, but not least, is the fact that a systems development effort conducted with meaningful involvement from user management is a much more gratifying and less nerve racking experience for the associated systems analysts and programmers than the kind of experiences they now have. This process for systems development will have a material effect on an organization's ability to hire and retain good systems personnel.

To review:

(1) Better system.

(2) Less maintenance.

(3) Greater chance for project approval.

(4) Less hectic.

(5) "No surprises."

(6) On-time conversion.

(7) Reduce resistance to change.

(8) Subsequent proprietary interest by users.

(9) Reduce turnover of systems analysts and programmers.

What I have just said comes from my own knowledge and experience. Less than two years ago I began teaching a three-day management seminar on systems development. Since then, over 900 top and middle user and systems management personnel have taken this seminar. As a result, various organizations are experiencing the benefits previously stated.

The following will highlight "Effective Data Systems Development," the process that incorporates meaningful user involvement in the development of their own data systems:

(1) From inception through conversion the systems development effort will be effected by a systems team composed of both user and systems personnel. The word is TEAM not committee. This means that a middle manager from the user department(s) will either be on a full-time or on a regular part-time assignment to the systems team.

(2) All team members, systems and users, are their department's representatives. They are responsible for keeping their co-middle managers and bosses informed of what the team is doing and relaying decisions and suggestions back to the team. Also, department management has a concomitant responsibility to communicate with their team representatives.

(3) Meaningful involvement of users extends to all users who will be affected by the new system. User team members are meaningfully involved when they do the kind of work on the team for which they are qualified due to their experience and knowledge of the user department's operation. User team members are not expected to perform sophisticated data processing tasks such as detail analysis, detail design and programming.

User management in the departments are meaningfully involved when they communicate with their team representatives, making decisions and offering suggestions.

(4) Responsibility for the success or failure of the systems development effort rests squarely with the team, not with the systems or user departments.

(5) A systems approach to the total systems development effort is taken by: 1.) Planning what will be done. 2.) Analyzing the present system. 3.) Detailing the new system. 4.) Implementing the new system, in that order.

a. Planning involves defining the problem(s), i.e., problem definition, and then determining what can be done to solve the problem(s), i.e., system objectives. Some refer to these two activities as a feasibility study.

b. The system being improved is then described in detail as to what it is presently doing and how it does it. The system's problems are also detailed.

c. The present system just described is then diagnosed as to what changes can reasonably be made to the present system to meet the system objectives. These changes are stated "in English" as quantified design guidelines.

d. The quantified design guidelines are given to detail designers (computer designers, analysts, programmers, and systems and procedures personnel) who detail and complete the design.

e. When the completed detail design is approved, the implementation phase begins. The computer systems personnel program and test the computer portion of the system and the user representatives train their own operating personnel. When both the systems testing and training are complete, the system is converted.

(6) The above noted tasks in a systems development effort are performed in the stated order; however, there is a measure of feedback in every systems effort. This is due mainly to mistakes and omissions on the part of the systems team, and those that help and communicate with them, and changes caused by factors which are both internal and external to the data system. Feedback has to be taken into account when determining the money, people and time required for the effort.

(7) A good problem definition is one that discovers all of the major problems, needs and requirements of the data system and the related requirements of other data systems that do or should interact with the data system being developed. Such a study may uncover more problems, needs and requirements than can be currently handled. Therefore, in defining the necessary solutions, decisions will be made as to what priorities to give to each requirement.

The important point here is that this initial effort all but eliminates the possibility of an existing requirement manifesting itself later on in the systems effort when it will cause disruption, delay and confusion. In addition, it may point up the need for more systems changes than previously thought. This can result in a larger initial effort or an incremental implementation of a larger effort stretched over a longer period of time.

(8) Not all of the solutions to data problems are found in automation. Most practical solutions contain combinations of two or more of the following: automation, reautomation, manual system changes, taking certain applications off the computer, policy changes, training, reorganization and improved supervision and operating controls.

(9) In the design phase the detail designers - some of whom are brought in to augment the systems members of the team - are working concurrently on the computer and manual portions of the system. This concurrence is important. Many poorly functioning automated data systems are designed first as computer systems and then, almost as an afterthought, consideration is given to the man/machine relationships of the user personnel who have to make the system work.

In this phase, the user members of the team are consulting the systems personnel on man/machine interface matters, working on detailing the policy changes and reorganization, and formulating their training program for operating personnel. The training program will be carried out during the implementation of the system.

(10) Whenever a quantified design guideline cannot be met without affecting another guideline, the detail designers must come to the systems team for the resolution of the trade-off. In this way, the systems team will decide all applicable trade-offs, thereby insuring that there will be "no surprises" for anyone.

(11) There is "open season on changes" from the time the systems development effort begins until the detail design is completed; however, once the completed design is approved for implementation, the design is frozen.

As noted before, throughout the planning, analysis and the design phases, the user representatives explain and show their co-managers and supervision what developments are being contemplated by the systems team and solicit their comments and suggestions. These suggestions and comments are carried back to the systems team for the team's consideration.

It is possible that an impasse can be reached when two or more participating departments - user vs. user or user vs. systems - cannot agree on a particular item. At this point the team leader will take the conflicting parties to higher management for a decision. Please appreciate that the whole Effective Data Systems Development procedure precludes many of the impasse-type situations that are now faced in systems development efforts. Also, if they do arise, they are handled and decided during the planning, analysis, and design phases, not during the programming effort, which is so often the case today.

(12) The completed design contains many things that are today considered program documentation. This means that the analysis and design tasks, much of which today is done during programming, are already completed at the time the detail design is approved. One of the jobs of the systems representative is to make sure that all of the detail decisions are made prior to the approval of the completed design. Before the systems department agrees to the completed design, they should feel that they have all of the information needed to "program in peace."

As you have probably noticed, I highlighted the phases of systems development prior to the programming, testing, training and conversion. This was done on purpose because it is during the planning, analysis and design phases that virtually all of the decisions are made and the majority of the total systems development time is spent. Under Effective Data Systems Development you could spend anywhere from 51 to 75 percent of the time properly preparing for the programming, testing, training and conversion with the remaining 25 to 49 percent of the time devoted to a smooth implementation.

I appreciate that many of you are not used to this kind of systems effort. Many of your current efforts consist of a little planning, a little analysis and a little design and a long

programming period. The reason for the long programming effort is that a majority of the programming time is not in programming but in analysis and design, or should I say redesign and reprogramming. Much of your present-day redesign and reprogramming carried on during the programming phase could be avoided if the proper people are meaningfully involved before programming rather than having to be involved during programming.

Therefore, to train management for MIS is to train them in systems development. Management must be convinced to meaningfully involve themselves in the development of their own MIS. They must appreciate that they cannot point their finger at their systems people and say "You do it!" It just does not work that way.

You realize that we are talking about a change in attitude. Let us effect this attitude change, get some decent management information systems developed and then I would like to return and give a talk entitled: "Training Management to Use Their MIS."

* * * * * *

The following may also be of interest:

Speech reprints (free)

"The Process of Effective Data Systems Development," presented at the 1968 Data Processing Management Association Convention.

"How to Unlock the Computer's Profit Potential," presented at the 1969 Spring Seminar of the Steel Valley Chapter of the Association for Systems Management.

Book ($15.00)

EFFECTIVE DATA SYSTEMS DEVELOPMENT

(over)

Seminar (3-day)

EFFECTIVE DATA SYSTEMS DEVELOPMENT

To obtain additional copies of this speech, the speeches or book noted above or information on the seminar, write or call:

Barnett Data Systems
Suite 507 - 1010 Rockville Pike
Rockville, Maryland 20852
Telephone: (301) 762-1288

HOW TO UNLOCK THE COMPUTER'S PROFIT POTENTIAL

Presented at the 1969
Spring Seminar of the
Steel Valley Chapter of the
Association for Systems Management

by

Arnold Barnett

Barnett Data Systems
Suite 507 - 1010 Rockville Pike
Rockville, Maryland 20852

HOW TO UNLOCK THE COMPUTER'S PROFIT POTENTIAL

Last year McKinsey & Company published a very enlightening research report entitled: "Unlocking the Computer's Profit Potential." This report was based on a survey of thirty-six large U. S. and European companies representing thirteen industries.

In essence, the report states that as far as technical achievement is concerned, the diversity and extent of computer applications are outrunning expectations; however, in terms of economic payoff, these current, technically superior applications are in real trouble. Costlier hardware, larger and increasingly costlier computer staffs are only part of the reason for the reduction of the profitability of each new computer application. This may sound like an example of the law of diminishing returns; however, not so because, in reality, the real profit potential of the computer has barely begun to be tapped. Therefore, something is wrong somewhere.

The McKinsey & Company report suggests that in the last few years the rules of the computer game have been changing but management's strategies have not changed. Not long ago management could leave the direction of the corporate computer effort largely in the hands of the data processing specialists. This was because of the nature of the systems being automated, mostly administrative and financial. No longer should this be the case; however, it is. Identification and selection of new computer applications are still predominantly in the hands of data processing specialists who, despite their technical proficiency, are poorly qualified to set the course of the corporate computer effort.

Speaking for myself now, this is generally true. However, management should not, by itself, set the course of the organization's computer effort. They should, in partnership with the data processing specialists, play a meaningful role in the decision-making process.

This sounds good; however, "management" is made up of individuals. How willing are individual managers to take over this vital function so necessary to help lead their organizations out of the land of diminishing returns into the realm of tapping the computer's real profit potential? Managers in most organizations are not willing to tackle this function. The prime reason being that they are not equipped to do so, and they know it.

Today there is a wide gap between management's ability and willingness to assume this responsibility and the willingness of the data processing specialists to accept more and more responsibility in this area. Therefore, somewhat by default, the computer specialists, alone, set the course of the organization's computer efforts.

This wide gap exists because management is ill prepared both in attitude and knowledge to assume this responsibility. Because of the lack of proper preparation, management feels secure in hiding behind their own jobs and telling their data processing specialists that it is they who are being paid to do systems work, not management. In addition, this lack of proper preparation means that new applications suggested by the data processing specialists are usually approved by management without the proper attention being paid to the systems planning and financial returns analysis, the two basic ingredients of a successful corporate computer program.

Why are managers so unprepared? Why are most managers unwilling to "grab the bull by the horns" and help lead their organizations into the realm of tapping the computer's real profit potential?

The main reason is that in the past, and continuing to this very day, management is being given either none or wrong education. Managers, whenever they can be gotten away to school, are being taught all about computer hardware and software and very, very little about such things as systems planning, cost/benefits analyses, systems evaluation, the process of systems development, their role in this process, and the specific techniques needed to carry out their roles. Management, rather than being taught something that they can effectively use, is being given a sales pitch on "what the computer can do for you." In addition, their class time is being taken up with demonstrations, detail programming exercises, and "hands on" experience designed to awe management with the power of the computer, understand the operation and "hardships" of the computer shop and be able to talk its language.

In all fairness, it should be said that the promulgators of this education achieve their goal. Today's management approves larger and larger expenditures for computers and computer staffs and, because of the detail training they received - which at the time was intellectually stimulating but of limited practical value - they do not know how and, therefore, do not wish to actively and meaningfully involve themselves in their own computer development projects. They do not realize that they can play a meaningful role without submerging themselves into the technical aspects of systems work.

To a certain degree, all of this is good for the data processing manager. As more and more computer specialists go it alone in directing their organization's

computer efforts, they are being promoted into positions such as "Vice President for Management Information" and other such elevated positions with the appropriate titles and salaries.

However, remembering the diminishing returns being experienced by most organizations in their computer development projects, the "control" of the organization's computer effort in the hands of computer specialists may prove for them to be disastrous. Currently the computer application developments in most organizations are beginning to be of the day-to-day operating and information systems variety. These are the most vulnerable too, because unlike the financial and administrative applications, which for the most part returned a savings in clerical help which more than offset the developing and operating costs, the current computer applications, especially the MIS systems, are not expected to return an operating profit. Their benefit to the organization is one of increasing management's efficiency, helping to create greater customer satisfaction, providing management with a better planning tool, etc. Therefore, if an implemented operating and/or information system fails to meet the requirements of the job, the financial consequences to the organization could be severe. Obviously, the person to get hurt in this situation would be the one who suggested the system and headed up its development.

An example of a development failure is the New York Stock Exchange's Central Accounting Service System; an effort of three year's time and three and one-half million dollars in development money, was scrapped. There were two consequences to this failure, not only was the development money lost but individual member brokers, tiring of the delays in the program, developed their own automated accounting systems. The time and effort spent by the brokers in developing their own automated accounting systems could have been used to improve their securities handling systems, thereby being better prepared for larger volumes of securities. Larger volumes did occur beginning last spring and, because of the lack of preparation, there were severe problems. For most organizations an analogous systems failure could well have an irreparable impact.

Data processing specialists should realize that even though they may be enjoying a new found stature, they are also approaching an abyss; the same aloofness that management is showing toward helping to direct the organization's computer program is also helping to contribute to the high cost of individual computer application developments and, in some cases, to partial or complete systems failure.

At this point, let us review:

1. Current, technically superior computer applications are not providing the economic payoff of former computer efforts; yet

the real profit potential of the computer has barely begun to be tapped.

2. The reason for the reduced economic payoff of current computer applications and the delay in tapping the real profit potential of the computer is that management has abrogated their share of direction of the organization's computer effort. Individually, managers are often unwilling and incorrectly trained to actively and meaningfully involve themselves in the development of their own automated systems.

3. Management, therefore, approves the suggestions of the data processing specialists and also confers upon them the burden of implementing their own suggestions. Recognizing this responsibility, many organizations have upgraded the position of the data processing manager, some right into the vice presidential level.

4. Herein may well lie the seeds for the downfall of the data processing specialist, because without management's active and meaningful participation in the development of their own operating and/or information systems, such sophisticated efforts stand to be, at best, unprofitable, a partial failure, or, at worst, a complete failure.

To begin to derive a practical answer to this problem, let us return to the McKinsey report. The following is a quote from page 31 of that report:

> "In almost every industry, at least one company can now be found that is pioneering in profitable new uses of computers. In such companies, our findings suggest, the key to success has been a strong thrust of constructive interest from corporate operating executives who have put their own staffs to work on computer development...
>
> "We believe that other companies will follow their lead. Indeed, it may soon be a nearly universal practice to transfer operating staff to computer development projects, either by making them members of a project team or by attaching them for a year or two to the corporate computer staff."

This passage tells us two things:

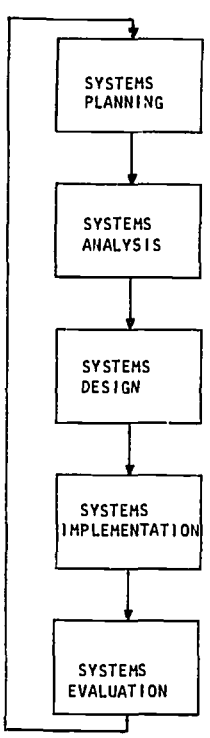
1. The direction of an organization's computer efforts need not be taken over completely by management; this should be a joint effort of both the data processing specialists and management.

2. As for each individual computer development project, its success rests, to a large extent, on the active and meaningful participation of operating management, i.e., representatives from the operating or staff departments for whom the computer system is being developed.

If this is true, and the experience of the more successful companies indicate that it is, then just how should an organization train its management so that they will be prepared and willing to actively and meaningfully participate in the development of their own data systems and be a partner with the data processing specialists in formulating the direction of their organization's computer efforts. How can operating executives be convinced that involvement in the development of data systems will not submerge them into the depths of detail technical activities? The answer lies in having a logical program for data systems development that maximizes the non-technical contributions of operating managers, and, also, can be both sold and taught to management in a relatively short period of time.

The remainder of this paper will highlight such a program. This program was not something thought up in response to the McKinsey report. It has existed since early 1967 and has been taught to over 500 persons from government and industry. In addition, the program is well documented in a book entitled: EFFECTIVE DATA SYSTEMS DEVELOPMENT.

The process of Effective Data Systems Development is depicted on the following page.



Admittedly, this format is the logical, classical approach to data systems development. However, its strength lies in the overall guiding systems principles of the effort and in the techniques of implementing of each of its phases.

First, let us review the guiding principles of the process of Effective Data Systems Development.

1. Operating management, the persons for whom the system is being developed, will actively and meaningfully involve themselves in each of the five phases of their system's development effort. The resulting benefits will be:

   a. Top management will be more likely to approve and support a "joint" systems effort rather than a systems effort proposed by either the data processing department or the operating department(s).

   b. The resulting system will be better because the persons for whom the system is being developed will be actively involved in such activities as defining requirements, describing the present system, diagnosing the present system, and designing the new system. Many of the problems that could arise during the programming, conversion, and system operation are being precluded by the involvement of the operating personnel during these initial phases.

   c. There will be "no surprises" for the operating departments when they review the final design of the new system. This is because their people will have been active in the total effort, beginning with the study of the problems of the current system through the design of the new system.

   d. The implementation of the new system will be smoother because the operating people who have had a substantial role in its development will consider it their own system. This will also have a psychological impact on the personnel of the operating department(s). They will think of the new system as one designed not by the data processing department but by their own management, a big factor in reducing their resistance to change.

2. The activities involved in each of these phases will be carried out in a sequential fashion, e.g., systems analysis will come before systems design; they will not be done at the same time.

3. Feedback is recognized as an integral part of this process necessitated mainly to rectify mistakes and omissions and adapt to changes affecting the system being developed.

4. Systems development efforts should be undertaken because there is a current and/or future <u>need</u> for improving an operation not because someone wants to use a computer.

5. When determining the improvements to be made to a system, a requirements rather than a capabilities approach should be taken. Requirements thinking places initial stress on the system's needs; whereas, capabilities thinking stresses the known resources available to fulfill these needs.

   For example: Requirements thinking will initially project all of the system's needs without regard to the organization's capability to fulfill these needs; whereas, capabilities thinking will concentrate on what can be done to help the system in the three hours of computer processing time available each day. The point is that even though the system developers who use the requirements approach will be forced to back off to the three hours of computer time, the resulting overall system improvements will be much better.

6. Last, and possibly the most important overall guiding principle, a <u>system</u> is being developed, not an application for the computer. This means that the system's developers will be looking in at the computer from the system, not looking out at the system from the computer.

Let us now examine each of the phases of activity in a systems development effort:

Systems Planning

In the systems planning phase a team of operating people and systems people study the problems and needs of a particular system. Their initial effort is to prepare a problem definition. The problem definition will lead into a study of the possible solutions to these problems and a set of system objectives are formulated. Key to the development and extent of the system objectives should be comprehensive cost/benefit analyses with dollar values "attached" to the benefits.

When top management reviews the system objectives with supporting cost/benefit analyses, they will give a "green light" to the systems development effort. At this point a formal systems team should be established. This team will be composed of both the data processing specialists and management-level operating personnel from the departments for whom the system is being developed. It should be made clear from the beginning that this is a team, not a committee, and it will function as such. If the project is large enough, team membership may be a full-time activity. The nucleus of the team is small. When required, the systems team will call upon systems analysts, programmers, internal auditors, operation researchers, training specialists, and others to perform the required detail work or act as consultants.

Systems Analysis

The systems analysis phase immediately follows the planning phase and consists of formally describing the present system, diagnosing the present system, and developing guidelines for the detailed systems design. In each of these activities, the operating people are heavily involved.

A good way to describe any data system is to work from the top down and depict each level of activity in successively increasing detail. This method has the advantage of allowing the operating members of the systems team to prepare the higher-level descriptions and to use systems analysts to do the more detailed lower-level descriptions. Two advantages accrue to the systems

development effort because of the use of operating people to describe their own system:

1. They know the present system and its nuances because they have worked in it for a period of time. A systems analyst from the data processing department could never gain this type of knowledge in the relatively short period of time he would have to spend analyzing it, especially as an outsider.

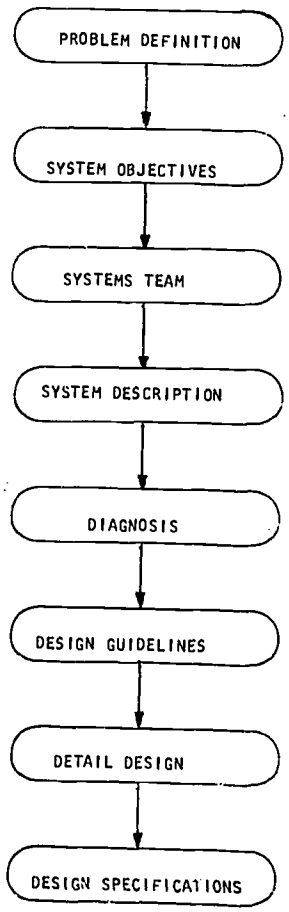
2. As the more detailed levels of the systems description are being delivered to the systems team by the systems analysts, the functional people will review them for reasonableness and accuracy and will tie them into the appropriate higher-level charts.

When the systems description documentation is finished, the systems team will use it to diagnose the present system. This diagnosis will, in essence, determine what changes can reasonably be made to the present system to meet the system objectives. This diagnosis will be done in light of the current state-of-the-art in computer technology and programming. From this activity will come a set of guidelines to the systems designers, which can also be considered a redefinition and detailing of the system objectives.

Systems Design

The detail systems designers will accept the design guidelines and begin the detail design. The designers will design the system the way the systems team performed the system description, in levels, from the top down. At any point in the design effort where the detail designers feel they cannot meet a guideline, the systems team will be notified to decide the appropriate trade-off. By making the designers clear all major trade-off decisions with the systems team, the functional people on the team will maintain their active participation and there will be "no surprises" for anyone when the design is finished. The functional people will be well qualified to do this because of their major role in describing and diagnosing the present system and in developing the guidelines for the designers. The product of the detail design is the design specifications.

In review, the systems development effort appears as follows:



Implementation Phase

The implementation phase consists of selecting equipment, if the system requires a new computer; programming, testing and debugging; training of the functional staff as to their role in the new system, and conversion from the existing system to the new system. In this phase, the operating people are most heavily involved in the training of their own staffs. It is they who should train their own people because they will teach the system in the language that the people will understand. Also, because of their own active and meaningful involvement in the system's development, they will sell the system. The mood of the functional people receiving the instruction will be more receptive because one of their own has materially helped in the system's development and it is he explaining the system, not some outsider.

Evaluation Phase

Immediately after the conversion, a formal evaluation scheme should be in operation to measure the operating performance of the new system. Performance criteria, those that are most meaningful for evaluation, are selected from the design guidelines. These criteria will be quantified and a range of acceptable values established. Operating performance which fall outside this range will be investigated for cause. This formal evaluation scheme has the following advantages:

1. By determining a range of acceptable performance, an organization is, at the same time, determining the levels of performance that, if reached, will be grounds for investigation as to their cause. This pre-thought as to what level of performance is worthy of investigation is a major step toward reducing over-reaction to that which, at first, may appear to be a major deviation. Sometimes, the over-reaction to what appears to be a problem does more harm than the performance deviation itself.

2. By charting actual system performance as reported in the evaluation data, the trends or direction of the system will be discernable. This will often lead to some short-range systems work and, in addition, will be an input to the long-range systems planning procedure.

In summary, this paper has pointed out:

1. The need for and the desirability of unlocking the computer's profit potential.

2. For an organization to achieve this, and possibly to save itself from a systems development fiasco, they must actively and meaningfully involve their operating and staff management in the development of their own systems. In addition, management should work in conjunction with the data processing specialists to chart the future course of the organization's "total" computer program.

3. However, no matter how good this sounds, we are dealing with individual managers and, for best results, they must be properly sold and (re)educated so that they will willingly involve themselves in such efforts. This goal can be accomplished if an organization employs a program of systems development that maximizes the contribution of the operating managers without wallowing them in detail and, accordingly, educates them as to the techniques of this program. With management actively and meaningfully involved, the economic payoff from current and future computer applications should rise substantially and, of equal impact, the probability of systems development fiascos will be substantially reduced.

* * * * *

THE PROCESS OF EFFECTIVE DATA SYSTEMS DEVELOPMENT

Presented at the 1968

Data Processing Management Association Convention

by

Arnold Barnett

Barnett Data Systems
Suite 507 - 1010 Rockville Pike
Rockville, Maryland 20852
Telephone: (301) 762-1288

Today, many organizations are spending substantial sums of money for data processing equipment and for the analysis and programming that go along with this equipment. These organizations are doing a wide variety of operational tasks with the computer which are much more significant than the initial routine financial and administrative applications. This trend toward computer applications which are involved in the day-to-day operation of the organization is manifesting itself into what is commonly called the "Management Information System."

However, if Management Information Systems are labeled "this generation," their development is being performed the same way it was two generations ago. When an organization decides that it wants an activity computerized, it writes the computer programs, makes the conversion and then begins the automated operation. The word "operation" in this case covers a multitude of sins. Usually the development of the application was quite hectic. There are many changes during the programming, a last-minute decision as to whether to slip the conversion date, computer installation problems, etc., etc., etc.

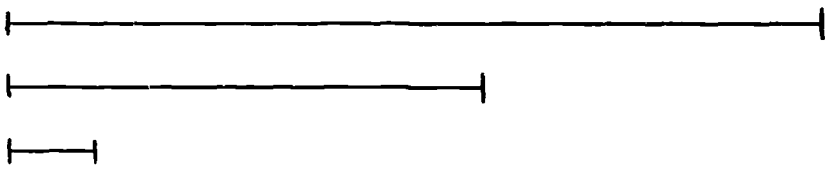
Finally, the system goes on the air only to be seriously challenged by the department for whom it was developed. After much patching and rewriting of programs, the application settles down to something that most everyone will live with. The final result is usually a system that falls short of what the functional department thought they were going to get.

The trouble is that computer system development efforts have been performed this way for so long that most people have conditioned themselves to it and they believe that nothing can be done about it. When hindsight is applied to particular application efforts, the conclusion is usually that "too little time was spent at the beginning and too much time at the end." Lessons are not learned, however, and the next application proceeds like the one before it.

If these troubles are "inherent" in the current administrative and operating applications, what lies ahead when organizations will be developing their Management Information Systems? Administrative and operating applications have a manual or semi-automated precedent system in existence and, at the very least, a straight conversion of this system can be made to the computer. In addition, success of the effort can be measured by how many dollars it saved or cost the organization.

However, with Management Information Systems, there is no comparable existing system to copy, nor is the system expected to save money for the organization. Management is buying a system that is going to cost more money because, in return, they expect the benefits of greater management efficiency, more customer satisfaction, a better planning tool, etc. Therefore, organizations that are moving toward a "Management Information System" should give serious thought to what can be done about their systems development procedures before large sums of money, time, and effort are wasted.

Another way of stating the current problem in systems development is to give labels to the following lines:

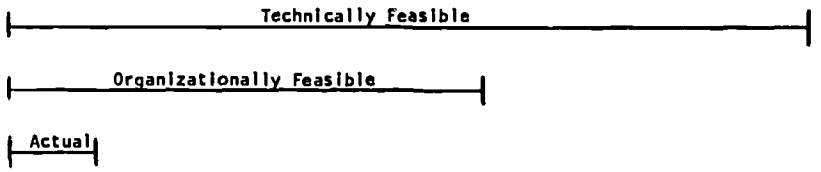


The first line stands for what can be accomplished with currently available computer technology.

The second line stands for what automation can be accomplished by an organization after accounting for the constraints of time, money, people, space, organization, politics, etc.

Sad to say, the third line represents what is actually accomplished.

Labeled, the lines look as follows:



A third way of looking at the current problem of systems development is to realize that the sophisticated operating and management systems, which are presently being worked on or are being contemplated for the near future, are extremely vital to the important day-to-day operations of the company. For operating management not to take an active and meaningful role in the development of these systems is tantamount to inviting a total waste of time and money or a system which, to their thinking, will be ill conceived, ill defined, poorly designed, hard to implement, and difficult to operate with. If the developed system is both "big and bad," the financial consequences to the organizations could be severe. An example of this is the New York Stock Exchange's Central Accounting Service System. The efforts of three years time and three and one-half million dollars in development money were completely scrapped.

Why do organizations continue to implement 3rd generation systems with first generation systems development procedures?

There are two basic reasons:

> One reason is that data processing specialists are too equipment and programming oriented. They tend to look at each application from the computer out, rather than the other way around. They do not give enough attention to the computer application as a portion of a larger system, a system to which the computer is giving a service.
>
> The second, and by far the more fundamental, reason for systems development troubles is that the management of the departments - the functional people - for whom the system is being developed do not actively and meaningfully involve themselves in the development of their own system.

There is no immediate solution to this problem; however, material improvements can be made. When these improvements are made organizations begin to see a greater return on their investment in data processing and the line on page 2 which represents "actual" accomplishment is substantially lengthened.

The path to improved systems development efforts lie in convincing middle and top management to employ a program for systems development that places initial emphasis on total system requirements and elicits their active and meaningful participation in the development of their own systems.

How does an organization implement such a program for systems development? As stated before, the biggest area for improvement in systems work lies with the persons for whom the systems are being designed. To gain their acceptance of a program which requires their active and meaningful involvement in developing their own systems, the following two-step procedure should be followed:

1. Make middle and top management aware of the reasons for their attitudes toward data processing, and

2. Give these same people an education that sells the idea of active and meaningful involvement - not hardware and software training - and, at the same time, gives them the tools to carry out their roles in a systems development effort.

The major reasons for a negative attitude toward data processing systems development efforts are:

1. Now, as in the past, operating management and staff specialists are effectively controlling their own operations. Their success depends on how well they combine technical proficiency in their

specialists with their political accumen and their ability to supervise and motivate their subordinates. Their productivity is, more often than not, determined by limits that they themselves set. They think that the computer, if brought into their operation, will change their time-honored way of doing things.

2. Data processing specialists and the persons they are trying to serve find it difficult to talk to each other. This is fostered by the physical separation of the Data Processing Department from the operating and staff offices, the sophistication of the computer, the complex programming needed to operate the computer, and the technical language used by systems people. It can be safely said that there is an "Understanding Gap" between the data processing specialists and the persons they are trying to serve.

3. Many potential data processing customers are concerned that if they automate a portion of their operation they will lose workers who are presently under their jurisdiction. They equate numbers of subordinates with their importance and fail to see how a computer, which takes away people, can add to their stature within the organization.

4. Managers attend formal ADP training programs. As computer orientation courses they are good. With regard to explaining the roles that managers play in systems development efforts, they frequently do more harm than good. Many managers return "fired up" about the potential of the computer but possess little knowledge of the systems considerations necessary to plan for and implement this "hardware."

   Also, these courses teach managers how to program the computer, how to add and subtract in binary and how many characters of data can be put on one inch of magnetic tape. The result is that the managers fear that active participation in a systems development effort will involve them in this low level of detail.

5. Some top executives have the idea that all that is needed to automate a system is a computer and some sharp programmers. When an automation effort fails, they tend to blame the Data Processing Department, either ignoring or not realizing the negative effect of their own and the operating department manager's non-involvement in the systems development effort.

After making management aware of the reasons for their attitude, proper management education should follow. The education that changes the attitudes of management toward meaningful involvement in systems development efforts focuses on three interrelated objectives:

1. The total process of systems development is illustrated. This opens management's eyes to the breadth of a systems development effort and to the myriad of tasks that must be addressed and successfully completed.

2. The respective roles of the data processing specialists and the functional people are explained. The students learn that there are some systems development tasks they can accomplish by themselves, there are many tasks in which they will work with the data processing specialists, and some detail or technically oriented tasks are strictly within the province of the data processing specialists.

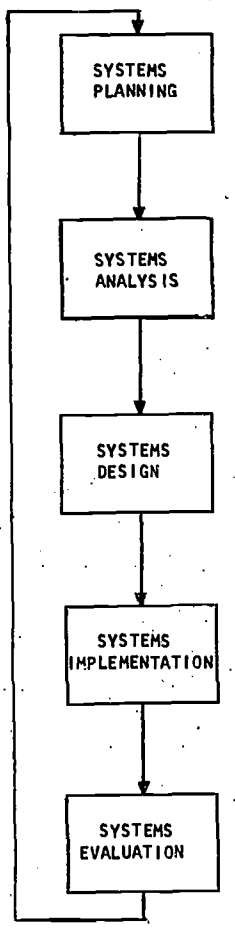
3. They will discover that they can perform "meaningful systems work" without becoming professional systems analysts. This is possible because the techniques of this procedure are geared to allow for maximum meaningful participation by the functional people working as a team with the data processing specialists.

This discussion will now highlight a systems development program that reduces the data processing specialist's attention to equipment and programming and increases the meaningful involvement of the persons for whom the system is being developed. Many of the concepts and procedures in this program can be adopted on an individual or group basis and profitably fitted into an organization's existing systems development program. It is of secondary importance whether or not all of the techniques discussed in this program are used. However, as more of the total program is adopted, the better will be the organization's systems development efforts.

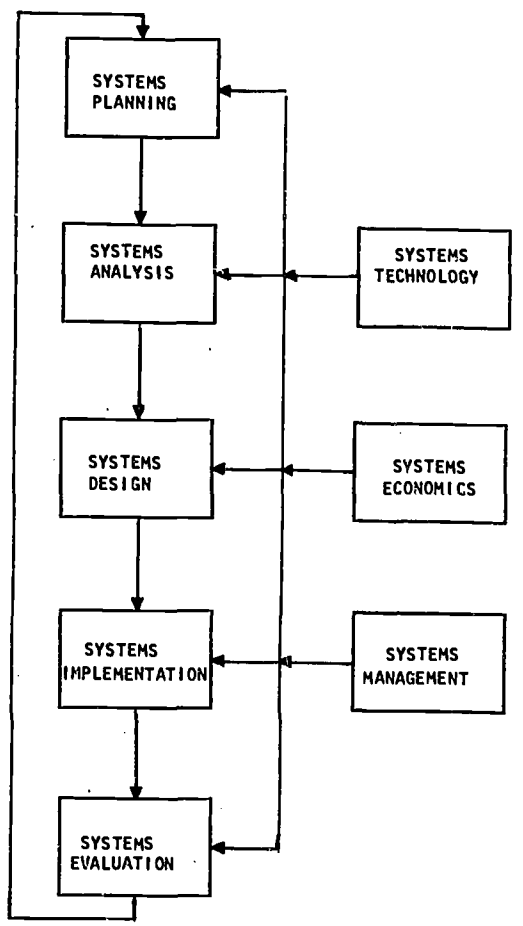
The Systems Development Process

A systems development effort is divided into five sequential phases of activity and three major functional activities.

The five sequential phases of activity are:



The three major functional activities are:



The following is a brief explanation of each of the systems development phases and functions.

During the Systems Planning Phase, the organization's data or information "problems" are defined, studied, solutions considered, systems development objectives formulated, and personnel assigned to work on fulfilling the objectives.

In the Systems Analysis Phase, the present system is documented, diagnosed to discover its weaknesses, and guidelines for the new system design are formulated.

The Systems Design Phase consists of taking the design guidelines - from the Analysis Phase - and using them as a basis for "constructing" the new system.

After the designed system has been approved, the Systems Implementation Phase will see new equipment selected and installed (if applicable), computer programs written and debugged, system operators trained, and the conversion made to the new system.

When the system begins operating, a formal evaluation procedure is put into effect to monitor the results of the operation. The Systems Evaluation Phase continues for the life of the system, materially aiding in the discovery of both system weaknesses and trends in processing activity. The results of these formal periodic evaluations are used in the Long-Range Planning for the subsequent system. This latter use of the Systems Evaluation is depicted in the feedback line from Systems Evaluation to Systems Planning.

Systems Technology concerns the equipment available for processing and transmitting data and the computer programming necessary to operate this equipment.

Systems Economics involves the determination of costs for the development and subsequent operation of a system and the management analysis for determining the "worth" of a particular system.

Systems Management concerns the day-by-day management and control of the systems development effort and handling resistance to change at all levels of the organization.

The Roles of the Participants and the Techniques of a Systems Development Effort

Systems Planning

Systems development efforts are begun for various reasons such as current data problems, the realization of future problems unless something is done now, and the advent of a new technology that makes a particular application feasible. When a problem is generally recognized as one worthy of serious consideration, a study team should be made up of data processing specialists and the functional personnel from the area(s) under study. The team should then formulate the system objectives and present them to top management for their concurrence.

In the context of this discussion, top management is relative to the scope of the problem. It could be the corporation's president and senior vice president if the problem were of the scope of a total management information system. If the scope of the problem does not exceed the boundaries of a plant, the plant manager would be "top management."

Note that we have begun the active and meaningful participation of the persons who will be the recipients of the new system. This involvement will continue through all of the phases of the systems development effort with the following benefits:

1. Top management will be more likely to approve and support a "joint" systems effort rather than a systems effort proposed by either the Data Processing Department or the operating department(s).

2. The resulting system will be better because the persons for whom the system is being developed are actively involved in such activities as defining requirements, describing the present system, diagnosing the present system, and designing the new system. Many of the problems that could arise during the programming, conversion, and system operation are being precluded by the involvement of the functional personnel during these initial phases.

3. The implementation of the new system will be smoother because the functional people who have had a substantial role in its development will

consider it their own system. This will also have a psychological impact on the personnel of the operating department(s). They will think of the new system as one designed not by the Data Processing Department but by their own management, a big factor in reducing their resistance to change.

When top management agrees to the system objectives and gives a "green light" to the systems development effort, a Systems Team should be established. This team will be composed of both the data processing specialists and management level operating personnel from the departments for whom the system is being developed. It should be made clear from the beginning that this is a team, not a committee, and it will function as such. If the project is large enough, team membership may be a full-time activity. The nucleus of the team is small. When required, the Systems Team will call upon systems analysts, programmers, and others to perform the required detail work. When appropriate, an internal auditor should also be a member of the Systems Team.

In review, active and meaningful involvement by operating management in their own systems development effort can begin with the "discovery" and definition of system problems. A study effort will lead to tentative solutions which will be followed by the formulation of system objectives. Top management will then approve the systems development effort, the proposal for which came from a joint effort by data processing and functional personnel. After approval to proceed has been given, a Systems Team will be established.

Systems Analysis

The Systems Analysis Phase immediately follows the Planning Phase and consists of formally describing the present system, diagnosing the present system, and developing guidelines for the detailed systems design. In each of these activities, the functional people are heavily involved.

A good way to describe any data system is to work from the top down and depict each level of activity in successively increasing detail. This method has the advantage of allowing the functional members of the Systems Team to prepare the higher level descriptions and to use systems analysts to do the more detailed lower level descriptions. Two advantages accrue to the systems development effort because of the use of functional people to describe their own system:

1. They know the present system and its nuances

because they have worked in it for a period of time. A systems analyst from the Data Processing Department could never gain this type of knowledge in the relatively short period of time he would have to spend analyzing it, especially as an outsider.

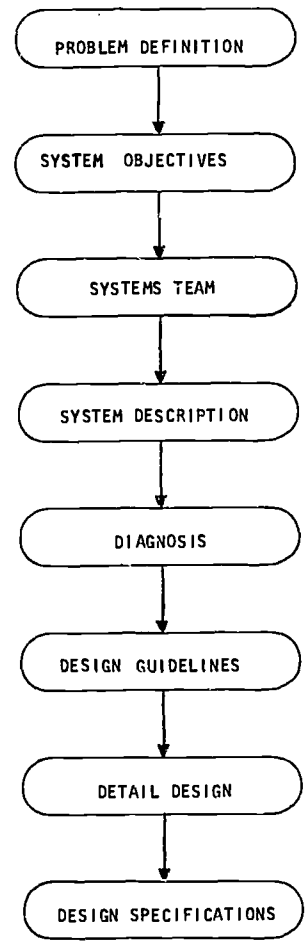
2. As the more detailed levels of the systems description are being delivered to the Systems Team by the systems analysts, the functional people will review them for reasonableness and accuracy and will tie them into the appropriate higher level charts.

When the systems description documentation is finished, the Systems Team will use it to diagnose the present system. This diagnosis will, in essence, determine what changes can reasonably be made to the present system to meet the system objectives. This diagnosis will be done in light of the current state-of-the-art in computer technology and programming. From this activity will come a set of guidelines to the systems designers, which can also be considered a redefinition and detailing of the systems objectives.

Systems Design

The detail systems designers will accept the design guidelines and begin the detail design. The designers will design the system the way the System Team performed the system description, in levels, from the top down. At any point in the design effort where the detail designers feel that they cannot meet a guideline, the Systems Team will be notified to decide the appropriate trade-off. By making the designers clear all major trade-off decisions with the Systems Team, the functional people on the team will maintain their active participation and there will be no surprises for anyone when the design is finished. The functional people will be well qualified to do this because of their major role in describing and diagnosing the present system and in developing the guidelines for the designers. The product of the detail design effort is the Design Specification.

In review, the Systems Development Effort appears as follows:



The next step in the process is the implementation of the design specifications. However, before that is discussed, note that there is no serious consideration of equipment or programming systems until the guidelines to the system design are formulated. This is done purposely to focus initial attention on the System requirements not the ADP requirements. The benefit is that with the kind of design specifications this procedure would produce an organization is much better prepared to intelligently select the proper equipment and programming to meet its systems needs.

In addition, this procedure calls for relatively more time being spent on the planning, analysis and design phases than is usually done today. The benefits of this are that by spending more time at "the beginning," less time is spent during the implementation of the system; the selection of the equipment will be better; and, when the system begins operation it will be much more acceptable and workable.

Implementation Phase

The Implementation Phase consists of selecting equipment, if the system requires a new computer; programming, testing and debugging; training of the functional staff as to their role in the new system, and conversion from the existing system to the new system. In this phase, the functional people are most heavily involved in the training of their own staffs. It is they who should train their own people because they will teach the system in the language that the people will understand. Also, because of their own active and meaningful involvement in the system's development, they will sell the system. The mood of the functional people will be more receptive because one of their own has materially helped in its development and it is he explaining the system, not some outsider.

Evaluation Phase

Immediately after the conversion, a formal evaluation scheme should be in operation to measure the operating performance of the new system. Performance criteria, those that are most meaningful for evaluation, are selected. These criteria will be quantified and a range of acceptable values established. Operating performance which falls outside of this range will be investigated for cause. This formal evaluation scheme has the following advantages:

1. By determining a range of acceptable performance, an organization is, at the same time, determining the levels of performance that, if reached, will be grounds for investigation as to their cause. This prethought

as to what level of performance is worthy of investigation is a major step toward reducing over-reactions to that which first appears to be a major deviation. Sometimes, the over-reaction to that which appears to be a problem does more harm than the performance deviation itself.

2. By charting actual system performance as reported in the evaluation data the trends or direction of the system will be discernable. This will often lead to some short-range systems work and, in addition, will be an input to the long-range systems planning procedure.

This discussion will not describe in detail the three functional areas of technology, economics and management; however, this knowledge is important to functional management and in the educational program they are amply covered.

In summary, this discussion has pointed out:

1. The pressing requirement for better "systems development efforts."

2. The need for:

   a) Data processing specialists to become more "systems" oriented as opposed to ADP oriented.

   b) The active and meaningful involvement of the persons for whom the systems are being developed.

3. A two-step procedure to gain this involvement:

   a) Make management aware of their attitude toward systems development.

   b) Via education, convince middle and top management of their role in the process of systems development. This is best done by teaching: 1) The total process of systems development. 2) Management's role in this process. 3) The techniques they will need to carry out their roles.

4. The techniques and procedures discussed are considered to be the best for doing systems development work with functional people actively and meaningfully involved. However, acceptance of a specific technique or procedure is secondary to the main goal of convincing management of their vital role in systems development.

5. Lest we forget, this educational program calls for the functional people to work with the data processing specialists on a Systems Team. This is much better than having functional people learn about data processing only to enable them to understand what the data processing specialists are saying and doing.

* * * * *

For additional copies of this reprint or for information about the 3-day course EFFECTIVE DATA SYSTEMS DEVELOPMENT, write or call:

Barnett Data Systems
Suite 507 - 1010 Rockville Pike
Rockville, Maryland 20852
Telephone: (301) 762-1288