Planning data processing education to meet job requirements

by JAMES O. HAMMOND
IBM Corporation
White Plains, New York

The greatest single expenditure that a data processing organization has today is for its human resource. This has been dramatically illustrated in several well-known studies. The cost of human resources continues to rise in data processing. Computers and systems are becoming more and more complex. It has become extremely difficult for programmers and analysts to know all the essential things about a system. This is especially true with the large, widespread teleprocessing networks. This complexity of systems is forcing more and more specialization in computer technology. At the same time, small computer staffs find that although a few staff members may know all that is required to support a system, it is frequently difficult to identify job responsibilities in such a way that maximum efficiency can be attained for the benefit of the business.

Most data processing organizations have major effects on their parent company’s ability to meet the needs of its business. It becomes of vital importance, therefore, that the data processing or information systems department, as they are frequently called, be managed as efficiently as possible providing the business support to its parent company in the most optimum fashion. Obviously, the management of any data processing department is an important key factor in its success. In this day and age of national economic lull and austerity moves in business to get more for the dollar spent, it has become even more important that management exercise its very best judgment at all times to run the data processing department more as a business and not just as a support function. In other words, have as valid justification that the money spent is worth what is being gained from the output.

There is another extremely important factor to consider in maintaining a high efficiency level in a data processing organization. This factor is the utilization of the staff employees or the bulk of the human resource. If the data processing staff is selected and trained to meet the requirements and needs of the business of the parent company, then a data processing organization has gone a long way toward solving efficiency problems not to mention personnel problems and potential large cost savings that can be gained.

The staff human resource portion of a data processing organization is the one addressed in this paper. More specifically, ways of defining and structuring data processing skills and job responsibilities are explained. Approaches are included on ways of defining an education program to provide the knowledge and skills that may not currently exist in the skills mix of the data processing staff.

Keeping in mind that all the other key factors of operating an efficient, productive data processing organization have not been forgotten, concentrate for a moment on how we might go about putting some form and rationale to human resource selection and training.

One of the most effective ways of relating data processing skills requirements, and in turn DP education, to a data processing organization is to start with the process being used to develop and implement data processing products. Relating skill requirements to the process is especially effective because the process is normally a well established one with which most company management are familiar both inside and outside the DP organization. Since the DP process is the mainstream around which the majority of all other DP activities in a department revolve, it is most likely a well defined and valid process. If the DP department is just being organized or for some other reason the process to be used is not well defined, then first things should be first—the process should be clearly resolved and defined.

The major steps of the DP process will vary from one DP organization to another and, of course, from
one company to another. This is expected and even desired if the organization is being primarily tailored to meet the needs of the parent company's business.

Figure 1 shows a selection of DP processes chosen from the DP organization of several companies in the United States. It can be seen, as was just mentioned, that the steps in the processes do vary. It is interesting, however, to notice that in spite of variations in the steps of the process and that some of the steps are called by different names with essentially the same meaning, that there is a thread of commonality through all the processes. There are some process steps that are common to all data processing organizations regardless of size, complexity, systems types, or structure. Three of these common steps are design, install, and maintain.

It is important that the major steps of the process be clearly defined in any given DP organization where the human resources are being studied. Once this is done a framework is established around which the human resource requirements can be structured. The process shown in Figure 2 has been selected for the purpose of illustrating how we can go about relating the human resource requirements to a DP process.

Considering the common thread through the processes again—design, install, and maintain—these general terms can now be expanded into the selected process.

Before we go further in skills requirements structuring, we must make sure that each step of the process is defined well enough so that there is no miscommunication. I have attached the following definitions to each step of this process. Keep in mind that there is nothing fixed about the process and the associated definitions that are being used in this example. Although, according to surveys made, this is one of the more common processes, the steps and definitions may be different for various DP organizations.

So that we may continue our analysis, here are the definitions that will be used.

**Process Steps**

I. Applications Analysis
II. Design Synthesis
III. Feasibility Validation
IV. System Design
V. Development and Testing
VI. Installation
VII. Operation
VIII. Maintenance

**Definitions**

I. **Applications Analysis**

This first step includes defining the problem, analyzing and documenting the existing system, describing the processing, defining output and performance requirements, estimating the value of what is required, and developing and present-
II. Design Synthesis

This step includes establishing project planning and control, checking and refining the systems requirements and initial specifications, developing preliminary design alternatives and selecting a solution that is most economically feasible and best meets the requirements specified (with supporting analysis from Feasibility Validation). The system solution is then proposed to management where a decision must be made whether to proceed into System Design or consider alternative solutions.

III. Feasibility Validation

This includes the processing testing the performance, technical feasibility and validity of design alternatives or existing solutions by creating prototypes, models or through simulation. Major cost and performance improvements can be made possible by the application of standard solutions and generalized program designs already available. Adaptations of existing systems or generalized programs may be entirely adequate solutions; therefore, no further design work would be required.

IV. System Design

This step includes planning and designing the details of the systems solution using the output from Design Synthesis and the feasibility model, selecting hardware and software, developing the new procedures and test plan, and identifying each program.

V. Development and Testing

Here the process is completed and proved by translating the system and program specifications into machine-readable code, testing and debugging this code and documenting each program module.

VI. Installation

This step includes converting data files, transactions, software and hardware used by the existing system to meet the requirements of the new system.
TITLE:
SENIOR ASSOCIATE PROGRAMMER, DEVELOPMENT.

GENERAL CONCEPT:
AN ACTIVE MEMBER OF AN INTERNAL DATA PROCESSING SYSTEM APPLICATION TEAM, RESPONSIBLE FOR DEFINING PROBLEMS, PROPOSING SOLUTIONS, DEVELOPING AND TESTING THOSE SOLUTIONS.
HE IS QUALIFIED TO SPECIALIZE IN THE AREA OF PROGRAM MODULE DEVELOPMENT.

SPECIFIC RESPONSIBILITIES:
(1) THE PRECISE DEFINITION OF PROGRAM MODULES (INPUT, TRANSFORMATION, AND OUTPUT) WHICH WILL IMPLEMENT THE SYSTEM DESCRIBED IN SYSTEM DESIGN.
(2) THE DOCUMENTATION, FLOWCHARTING AND CODING OF THE MODULES SPECIFIED.

COMMUNICATES WITH:
TESTING SPECIALIST FOR RESULTS OF MODULE AND SYSTEM TESTS.
SYSTEM DESIGN SPECIALIST TO REQUEST MODIFICATION OF SYSTEM SPECIFICATIONS, IF THEY ARE FOUND UNCLEAR.
MANAGER TO REPORT PROGRESS AND RECEIVE ASSIGNMENTS.

APPROPRIATE TOOLS, TECHNIQUES AND SKILLS:
ADVANCED PL/1
MODULAR PROGRAM DESIGN
PROGRAMMING TECHNIQUES
OPERATING SYSTEMS INTERFACE

PROMOTION POSSIBILITIES:
STAFF PROGRAMMER, DEVELOPMENT AND TESTING
SENIOR ASSOCIATE PROGRAMMER, SYSTEM DESIGN

Figure 5—Job description

VII. Operation
This step includes running the designed and tested system (or application) on the computer.

VIII. Maintenance
This includes the maintenance of the software systems programs and applications programs, e.g., changes, extensions, and optimization.

There are specific skills, tools and techniques that must be known and applied to accomplish each of the steps in this process. If the organization is a large one, a staff member might specialize in one or two of these steps and be required to master a smaller number of tools, but be able to apply these with considerable expertise. A small DP organization would not have a requirement for such extensive expertise; the staff member may, therefore, be skilled enough to perform several of the steps in the process to a much less level of complexity.

At this point the DP organization functions and job responsibilities can be related to the process that has been described. The organization functions can be identified as being three main segments of activity:

1. Application Development
2. System Support
3. Operations

Figure 3 shows how these organizational functions can be related to the process. Notice the overlap among the functions. For example, the system support function overlaps the application development function in the development and test step and installation step of the process. In large installations where a high degree of specialization may exist, these overlaps indicate the points where responsibilities are passed from one person or group to another. It is extremely crucial that these key points of communication are identified and the responsible staff members are made aware that this “passing of the baton” so to speak, is an important part of their responsibility. It is equally important that they be adequately trained to carry out these tasks with great efficiency.

Now take another cut at the same process from the individual staff member point of view. For the purpose of analyzing how this might be done, consider the applications development function alone for a minute. The staff working within this function of a DP organization...
Responsibilities/Skills
(Be Able To:)

Application Analysis
- Analyze existing system/applications
- Document system
- Define system requirements
- Development and present initial specifications

Design Synthesis
- Refine initial specification
- Develop conceptual solutions—design alternatives
- Identify and select known/existing solutions
- Specify details of each design alternative
- Estimate development, conversion, installation and operating costs
- Test and evaluate alternatives and select best solution
- Present proposed system solution

Feasibility Validation
- Create and test prototype systems
- Construct and analyze or simulate models
- Compare prototype or model to required systems performance

Technical Strength Definitions
0 has no knowledge
1 understands capability and applicability
2 can use technique effectively
3 can maintain, modify and create new features
4 can synthesize and evaluate to reduce or eliminate analysis, machine runs, development or tests based on his wide knowledge of alternatives and depth of experience.

Systems Design
- Development systems design work plan
- Document system in detail
- Design forms and reports
- Finalize record and file requirements
- Define data editing and validation rules
- Select final hardware configuration and software
- Document controls, audit trails, back-up procedure
- Write new and revise old procedures
- Plan system test and prepare test data
- Prepare computer operations work flow
- Document computer flow and identify each program

Development and Testing
- Define program modules
- Flowchart, code and document program modules
- Design and implement program module tests
- Implement system tests specification in system design

Planning Data Processing Education is made up for the most part of one of three types of personnel: analysts, programmers, and managers. The analysts are known as systems, application or program in most cases. The analysts' job responsibilities can range across the analysis, synthesis, validation, and design of DP systems and applications. Programmers are generally of two primary types, applications and systems. Applications programmers program and test primarily business applications whereas system programmers program primarily control and system applications. There are a wide range of job responsibilities performed by programmers at all levels of complexity. The managers referred to here are technical managers. It is important to identify the job responsibilities of higher management along with all the other members of the organization but higher managements' job responsibilities are broader, less technical, and more administrative. For this reason they have been excluded in the sample being used in this paper. The technical manager is one who has direct supervision over a small staff of programmers or analysts and is able to understand technical detail, at least within his assigned responsibilities.

There are numerous titles for programmers and
RESPONSIBILITY (SKILLS)  
(BE ABLE TO:)

APPLICATION ANALYSIS  
ANALYZE EXISTING SYSTEM/APPLICATIONS  
DOCUMENT SYSTEM  
DEFINE SYSTEM REQUIREMENTS  
DEVELOP & PRESENT INITIAL SPECIFICATIONS

DESIGN SYNTHESIS  
REFINE INITIAL SPECIFICATION  
DEVELOP CONCEPTUAL SOLUTIONS—DESIGN ALTERNATIVES  
IDENTIFY AND SELECT KNOWN/EXISTING SOLUTIONS  
SPECIFY DETAILS OF EACH DESIGN ALTERNATIVE  
ESTIMATE DEVELOPMENT, CONVERSION, INSTALLATION AND OPERATION COSTS  
TEST AND EVALUATE ALTERNATIVES AND SELECT BEST SOLUTION  
PRESENT PROPOSED SYSTEM SOLUTION

FEASIBILITY VALIDATION  
CREATE AND TEST PROTOTYPE SYSTEMS  
CONSTRUCT AND ANALYZE OR SIMULATE MODELS  
COMPARE PROTOTYPE OR MODEL TO REQUIRED SYSTEMS PERFORMANCE

SYSTEM DESIGN  
DEVELOPMENT SYSTEM DESIGN WORK PLAN  
DOCUMENT SYSTEM IN DETAIL  
DESIGN FORMS AND REPORTS  
FINALIZE RECORD AND REQUIREMENTS  
DEFINE DATA EDITING AND VALIDATION RULES  
SELECT FINAL HARDWARE CONFIGURATION AND SOFTWARE  
DOCUMENT CONTROLS, AUDIT TRAILS BACK-UP PROCEDURE  
WRITE NEW AND REVISE OLD PROCEDURES  
PLAN SYSTEM TEST AND PREPARE TEST DATA  
PREPARE COMPUTER OPERATIONS WORK FLOW  
DOCUMENT COMPUTER FLOW AND IDENTIFY EACH PROGRAM

DEVELOPMENT AND TESTING  
DEFINE PROGRAM MODULES  
FLOWCHART, CODE AND DOCUMENT PROGRAM MODULES  
DESIGN AND IMPLEMENT PROGRAM MODULE TESTS  
IMPLEMENT SYSTEM TESTS SPECIFICATION IN SYSTEM DESIGN

<table>
<thead>
<tr>
<th></th>
<th>BASIC</th>
<th>INTERMEDIATE</th>
<th>ADVANCED</th>
</tr>
</thead>
<tbody>
<tr>
<td>APPLICATION ANALYSIS</td>
<td></td>
<td></td>
<td>0 1 2 3 4 4 4</td>
</tr>
<tr>
<td>DOCUMENT SYSTEM</td>
<td>0 2 3</td>
<td>4 4 4 4</td>
<td>0 2 3 4 4 4</td>
</tr>
<tr>
<td>DEFINE SYSTEM</td>
<td></td>
<td></td>
<td>0 2 3 4 4 4</td>
</tr>
<tr>
<td>REQUIREMENTS</td>
<td></td>
<td></td>
<td>0 2 3 4 4 4</td>
</tr>
<tr>
<td>DEVELOP &amp; PRESENT</td>
<td>0 2 3</td>
<td>4 4 4 4</td>
<td>0 2 3 4 4 4</td>
</tr>
<tr>
<td>INITIAL SPECIFICATIONS</td>
<td>1 2 3</td>
<td>4 4 4</td>
<td>1 2 3 4 4 4</td>
</tr>
<tr>
<td>DESIGN SYNTHESIS</td>
<td></td>
<td></td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>REFINE INITIAL</td>
<td>0 2 3</td>
<td>4 4 4 4</td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>SPECIFICATION</td>
<td></td>
<td></td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>DEVELOP CONCEPTUAL</td>
<td>0 2 3</td>
<td>4 4 4 4</td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>SOLUTIONS—DESIGN</td>
<td></td>
<td></td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>ALTERNATIVES</td>
<td></td>
<td></td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>IDENTIFY AND SELECT</td>
<td></td>
<td></td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>KNOWN/EXISTING</td>
<td></td>
<td></td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>SOLUTIONS</td>
<td></td>
<td></td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>SPECIFY DETAILS</td>
<td>0 2 3</td>
<td>4 4 4 4</td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>EACH DESIGN ALTERNATIVE</td>
<td></td>
<td></td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>ESTIMATE DEVELOPMENT</td>
<td></td>
<td></td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>CONVERSION, INSTALLATION AND OPERATION COSTS</td>
<td></td>
<td></td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>TEST AND EVALUATE ALTERNATIVES AND SELECT BEST SOLUTION</td>
<td></td>
<td></td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>PRESENT PROPOSED SYSTEM SOLUTION</td>
<td></td>
<td></td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>FEASIBILITY VALIDATION</td>
<td>0 2 3</td>
<td>4 4 4 4</td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>CREATE AND TEST</td>
<td></td>
<td></td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>PROTOTYPE SYSTEMS</td>
<td>0 2 3</td>
<td>4 4 4 4</td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>CONSTRUCT AND ANALYZE</td>
<td></td>
<td></td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>OR SIMULATE MODELS</td>
<td></td>
<td></td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>COMPARE PROTOTYPE OR</td>
<td>0 2 3</td>
<td>4 4 4 4</td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>MODEL TO REQUIRED</td>
<td></td>
<td></td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>SYSTEMS PERFORMANCE</td>
<td></td>
<td></td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>SYSTEM DESIGN</td>
<td>0 2 3</td>
<td>4 4 4 4</td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>DEVELOPMENT SYSTEM</td>
<td>0 2 3</td>
<td>4 4 4 4</td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>DESIGN WORK PLAN</td>
<td>0 2 3</td>
<td>4 4 4 4</td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>DOCUMENT SYSTEM</td>
<td>0 2 3</td>
<td>4 4 4 4</td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>IN DETAIL</td>
<td></td>
<td></td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>DESIGN FORMS AND</td>
<td>0 2 3</td>
<td>4 4 4 4</td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>REPORTS</td>
<td></td>
<td></td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>FINALIZE RECORD</td>
<td>0 2 3</td>
<td>4 4 4 4</td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>AND REQUIREMENTS</td>
<td></td>
<td></td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>DEFINE DATA EDITING</td>
<td>0 2 3</td>
<td>4 4 4 4</td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>AND VALIDATION RULES</td>
<td></td>
<td></td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>SELECT FINAL HARDWARE</td>
<td>0 2 3</td>
<td>4 4 4 4</td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>CONFIGURATION AND</td>
<td></td>
<td></td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>SOFTWARE</td>
<td>0 2 3</td>
<td>4 4 4 4</td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>DOCUMENT CONTROLS,</td>
<td>0 2 3</td>
<td>4 4 4 4</td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>AUDIT TRAILS BACK-UP</td>
<td>0 2 3</td>
<td>4 4 4 4</td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>PROCEDURE</td>
<td></td>
<td></td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>WRITE NEW AND REVISE</td>
<td>0 2 3</td>
<td>4 4 4 4</td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>OLD PROCEDURES</td>
<td></td>
<td></td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>PLAN SYSTEM TEST AND</td>
<td>0 2 3</td>
<td>4 4 4 4</td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>PREPARE TEST DATA</td>
<td></td>
<td></td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>PREPARE COMPUTER</td>
<td>0 2 3</td>
<td>4 4 4 4</td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>OPERATIONS WORK FLOW</td>
<td></td>
<td></td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>DOCUMENT COMPUTER</td>
<td>0 2 3</td>
<td>4 4 4 4</td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>FLOW AND IDENTIFY</td>
<td></td>
<td></td>
<td>0 2 3 4 4 4 4</td>
</tr>
<tr>
<td>EACH PROGRAM</td>
<td></td>
<td></td>
<td>0 2 3 4 4 4 4</td>
</tr>
</tbody>
</table>

Figure 7—Programmer/analyst in application development environment responsibility (skills) matrix

analysts. It is best, of course, to define job levels using the titles of a given DP organization. In Figure 4, the levels—basic, intermediate, and advanced,—have been used to relate job levels to the process. The bar graphs under these levels give an indication of the approximate average level required to perform the job responsibilities in the corresponding process steps. For example, most senior staff personnel have had experience and/or training in development and testing before advancing to more senior positions in either systems design, feasibility validation, or design synthesis. This type of structuring forms a rather neat array of potential career paths.

Job levels, titles, and major responsibilities are traditionally structured into a job description. Job descriptions can range in form from the one sheet summary shown in Figure 5 to very detailed descriptions. In all cases, it is worthwhile specifying the...
System Design

<table>
<thead>
<tr>
<th>Step</th>
<th>Basic</th>
<th>Intermediate</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Develop system design work plan</td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>Document system in detail</td>
<td>0</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>Design forms and reports</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>Finalize record and file requirements</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>Define data editing and validation rules</td>
<td>0</td>
<td>2</td>
<td>3</td>
</tr>
</tbody>
</table>

Select final hardware configuration and software | 0     | 2            | 3        |
Document controls, audit trails, back-up procedure | 0     | 1            | 2        |
Write new and revise old procedures              | 1     | 2            | 3        |
Plan system test and prepare test data          | 0     | 2            | 2        |
Prepare computer operations work flow            | 1     | 2            | 3        |
Document computer flow and identify each program | 1     | 2            | 3        |

Technical Strength Definitions

0 has no knowledge
1 understands capability and applicability
2 can use technique effectively
3 can maintain, modify, and create new features
4 can synthesize and evaluate to reduce or eliminate analysis, machine runs, development or tests based on his wide knowledge of alternatives and depth of experience.

Figure 8—Programmer/analysts—Responsibility matrix expanded

knowledge and skills required as well as promotion possibilities. This information is valuable to management both in utilizing the employees best talents and planning for his career advancement.

A further expansion is necessary of the knowledge and skills required to do the jobs within each step of the process. If some type of job analysis has not been performed on the data processing organization in question, then now is the time. What the job involves, the complexity of the technical level, the associated responsibilities, and, last but not least, the knowledge and skill level required to do the job all must be defined to fulfill the requirements specified in the steps of the process. Much of this information can be structured on the job description summary sheet that was just described.

Job analysis should be done by professionals who have had experience in making such analysis. This type of study is usually performed for a fee by an outside consultant. If the consultant is carefully chosen the results will be well worth the money but be sure that he has extensive data processing expertise on his staff to participate in the analysis. Also make sure that his analysis is going to be provided to you in a form that fits your needs and is tailored to the process in your organization.

There are many good approaches to job analysis. Most of them involve some type of general experience questionnaire and the observation and interview of selected employees doing their job. It is important to make sure that the employees have a good understanding of what is being done and why job-related questions are being asked. Employees quickly become skeptical and concerned when a lot of questions are asked about their job and how it is performed. The cooperation given by the staff in programs such as this will be much better and with a more positive attitude if the reasons are first clearly explained.

Under each of the process steps the major skills, actions, and knowledge are identified that are important in accomplishing the step in the DP organization. Try to refine these major items as much as possible so they will be concise yet descriptive of the responsibility. Sometimes it is difficult to express all the items purely in the terms of skills. For this reason be practical and realistic about the results by using a mixture of skills, actions, and knowledge. This approach seems to render a better level of definition.

Figure 6 shows how this expansion can be accomplished. Keep in mind that we are talking about the application development function of the DP organization and the associated programmer and analyst staff. For this reason the expansion shown is for the first five steps of the process.

The technical strength definitions below show how knowledge levels can be identified. Once again these definitions should be prepared with the requirements of the organization in mind. There must be an adequate
range in the knowledge levels. With programmers and analysts it is especially desirable to be able to identify whether they can explain a concept and apply it as well.

**Technical Strength Definitions**

0. has no knowledge  
1. understands capability and applicability  
2. can use technique effectively  
3. can maintain, modify and create new features  
4. can synthesize and evaluate to reduce or eliminate analysis, machine runs, development or tests based on his wide knowledge of alternatives and depth of experience.

You will notice that these levels are primarily for programmers and analysts so that we may continue to expand our definition of the application development function.

All the data that is needed to put together a summary skills matrix of a selected DP organization should now be available. The data includes:

1. Definition of the steps in the DP process to meet the needs of the business.  
2. Definitions of job responsibilities to meet the process requirements.  
3. Association of these job responsibilities with defined jobs.  
4. Identification of the level and possibly the title of each of the defined jobs.  
5. List of the major relevant skills, actions, and knowledge to meet the defined job responsibilities.  
6. Define the strength levels of the skills and knowledge requirements.

A summary skills matrix can now be constructed by inserting the strength indicators required for each defined activity of each process step. The strength indicators are placed in the appropriate job title or level column.

Figure 7 shows how the technical strength indicators are added to the System Design step of the process. The strength level definition and the job level to which it is associated is determined primarily from the data obtained as a result of the job analysis and survey of the staff's skills.

When the summary skills matrix is constructed for all members of the DP organization, it becomes an invaluable management tool in planning and obtaining the required skill mix. The skills matrix also acts as a key index in planning tailored education for the staff. This can lead to significant savings in education expenses to a company.

Using the summary skills matrix and data from the job analysis and knowledge surveys, a trained industrial educator can determine what additional knowledge and skills are needed to enhance that already possessed by the staff. The identification of the missing skills and knowledge is the foundation on which future training

**Name:**  
Business System Design

**Duration:**  
Five days

**Audience:**  
Those with the responsibility for the detailed design of a business system.

**Prerequisites:**  
Experience in program development and testing or successful completion of the development and testing curriculum.

**Objectives:**  
Upon successful completion of this course, the student will be able to:  
- Develop cost estimates and a work plan for scheduling and controlling the work through the design phase.  
- Define the record content, file organization, storage requirements and access methods.  
- Define data editing and validation rules, controls and audit trails.  
- Design the necessary forms and reports.  
- Complete the detailed documentation of the system, specifying the input, output, processing and file requirements of each program.  
- Prepare the systems test plan.

**Topics**

<table>
<thead>
<tr>
<th>Topic</th>
<th>Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Systems design cost estimating planning and scheduling</td>
<td>2</td>
</tr>
<tr>
<td>Detailed record design, file organization and access method specification</td>
<td>4</td>
</tr>
<tr>
<td>Editing, validation, controls and audit trails</td>
<td>2</td>
</tr>
<tr>
<td>Forms and reports design</td>
<td>2</td>
</tr>
<tr>
<td>Procedure writing</td>
<td>2</td>
</tr>
<tr>
<td>Systems test planning, test data and file preparation</td>
<td>4</td>
</tr>
<tr>
<td>Program specification and documentation</td>
<td>4</td>
</tr>
<tr>
<td>Estimating programming development time</td>
<td>2</td>
</tr>
<tr>
<td>Case study</td>
<td>8</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>30</strong></td>
</tr>
</tbody>
</table>

* These topics teach to the 2 and 3 technical strength levels.

Figure 9—Course specification
can be based. Much of this needed training can be accomplished on the job or through self-study. Some of it will require more formal classroom education. In any case, the education will be tailored toward the business needs and requirements since it is based on a foundation geared toward that same goal.

Regardless of whether the training program required is going to be developed in-house or contracted out to an education company or computer manufacturing company, it is good to have at least a general specification of the type of training needed.

Using the related job responsibilities boxed in on Figure 8, a training specification can be written. Job responsibilities that are related should be clustered together so that eventually a series of topics or modules can be defined to teach the relevant subjects. Related topics, in turn, are normally grouped together to form a course.

The course specification shown in Figure 9 is one designed to teach to the 2 and 3 strength level on all the job responsibilities boxed in Figure 8. Notice that the course specification identifies the most pertinent facts about the training needed. The audience, prerequisites and a clear definition of the objectives are the most important items. Individual topics that make up the course and the approximate duration of each can be included when they are needed. The purpose for using a duration on each topic is to give some indication of the depth that topic covers.

I said at the beginning of this paper that surveys show that the greatest expenditure that a data processing organization has is for its human resource. Actually it is the knowledge that is being purchased. This statement applies equally as well to most all human resource requirements in industry. In Peter F. Drucker's recent book, *The Age of Discontinuity*, he treats knowledge as one of the four discontinuities. Mr. Drucker says, "Knowledge, during the last few decades, has become the central capital, cost center and the crucial resource of the economy."

Since knowledge is a very valuable and expensive commodity it should be treated as such. Planned education according to the needs of a business will impart knowledge where and when it is required.