Analysis of Task Analysis Procedures

David H. Jonassen
School of Education
University of North Carolina at Greensboro
Greensboro, NC 27412-5001
and
Wallace H. Hannum
University of North Carolina at Chapel Hill

Abstract. For the new developer, deciding which task analysis procedures to use can be confusing. In this article, we describe the five functions comprising the task analysis process: inventorying, describing, selecting, sequencing, and analyzing tasks. We then describe some critical distinctions in the task analysis process: micro/macro level, top-down/bottom-up, and job/learning task analysis. We then combine the functions and distinctions in task analysis into a quasi-algorithm to suggest which of thirty task analysis procedures may be used to fulfill each of the functions. Those procedures are described briefly in the Appendix.

Introduction

This article is predicated on three assumptions:

1. Task analysis, regardless of how it is defined, is an integral part, probably the most integral part, of the instructional development process. All instructional development models to date include some task analysis procedures (Andrews & Goodson, 1980). Most developers indicate that a poorly executed task analysis will jeopardize the entire development process.

2. Task analysis may be the most ambiguous process in the development process. Task analysis represents one or more steps in the instructional development process, which purports to be a science; however, it contains uncertain knowledge and multiple interpretations. We contend that the ambiguity results from the diversity of procedures and definitions of the process. Definitions of task analysis range from the "breakdown of performance into detailed levels of specificity" to "front-end Analysis, description of mastery performance and criteria, breakdown of job tasks into steps, and the consideration of the potential worth of solving performance problems" (Harless, 1980, p. 7). This article evolved from the confusion experienced by an instructional design class trying to conceptualize the task analysis process. Trying to reconcile the myriad task analysis procedures performed at different levels in different situations can be exasperating. The option, too often practiced, is to use a single procedure that makes sense to the developer and apply it uniformly, thus overgeneralizing it to every instructional situation. Experienced instructional developers may know intuitively which procedures to apply in various settings. However, the neophyte's semantic network of task analysis constructs is not sufficiently developed to allow him to know "intuitively" when to apply different task analysis "scripts" (i.e., procedures). So clarification should help the beginning developer.

3. Recent reviews of task analysis (Foshay, 1983; Kennedy, Esquire, & Novak, 1983) have been useful in identifying the various task analysis procedures and their functions. However, simply knowing what tools are available will not rectify the confusion encountered by inexperienced developers. The confusion results from not knowing which task analysis procedures to use in various situations. Foshay (1983) made some useful recommendations about when to apply which model, but he reviewed only three out of a long list of potential task analysis procedures. What design students need is guidance on when and where to apply the various task analysis procedures.

This article is dedicated to that purpose. We do not intend to review each procedure comprehensively. Nor can we claim a foolproof algorithm for recommending which procedures to apply in all circumstances. Task analysis remains too inexact a science to accomplish that goal. In order to make suggestions about when to apply the various task analysis procedures, we first must clarify what functions are integral to the process. Then, we will briefly discuss some situational variables that affect the task analysis process. From those variables, we shall derive a quasi-algorithm for suggesting alternative task analysis procedures that may be used to accomplish each task analysis function. Those procedures are annotated in the Appendix. Our purpose is to provide a framework for selecting and understanding task analysis procedures and applying them to the task analysis process.

Task Analysis Functions

Much of the confusion about task analysis that frustrates inexperienced instructional developers results from a lack of agreement about what the process of task analysis involves. What exactly do designers do when they conduct a task analysis? That varies greatly among developers.

In some contexts, task analysis is limited to developing an inventory of steps routinely performed on a job. In others, task analysis is functionally synonymous with front end analysis, including all instructional development procedures prior to determining instructional strategies. According to Romiszowski (1981), task analysis procedures pervade the four levels of instructional design. At the course level (Level 1), overall objectives are defined. At the lesson level (Level 2), objectives are refined and sequenced, and entry level requirements are specified. At the instructional event level (Level 3), the detailed behaviors are classified. At the learning step level (Level 4), task statements are elaborated on, as individual steps in the task are identified. Each step of this top-down, macro-to-micro instructional design process is heavily dependent on task analysis.

Kennedy, Esquire, and Novak (1983) recently identified the different components of task analysis as occurring in two separate phases. The task description phase consists of identifying, refining and ordering tasks. According to their survey, the instructional phase consists of the processes of: (1) specifying goals, needs, and objectives; (b) developing analysis tools (such as taxonomies and learning hierarchies); and finally (c) identifying outcome specifications (such as product descriptions and training considerations). They found considerable disparity among instructional development models in terms of the components each included as part of the task analysis process. One model included two of the ten, while another included only eight. This disparity creates even more confusion for instructional developers and particularly for students. Just what does the task analysis process involve?

We contend that the task analysis process consists of five distinct functions: (a) Inventorying tasks, (b) Describing tasks, (c) Selecting tasks, (d) Sequencing tasks and task components, and (e) Analyzing task and content level. These are functional descriptions of what is included in the task analysis process. The task analysis process, as performed in different settings, may involve some or all of these functions. The combination of functions that are performed depends upon situational design variables to be discussed later. Each function may be accomplished by using different procedures (see Appendix). Yet each different procedure imposes constraints on that function. So care must be exercised in selecting a procedure for accomplishing each of the task analysis functions. The purpose of this article is to provide some selection criteria to assist the beginning developer in deciding which procedures can be used to accomplish each of the task analysis functions. Deciding which functions must be accomplished depends upon the nature of the task, the instructional situation, the outcomes required: and the experience of the developer.

> Task analysis is an integral part of the instructional development process. A poorly executed task analysis will jeopardize the entire development process.

Inventorying Tasks

Task inventory is the process of identifying the relevant tasks that may be considered for further instructional development. This inventory may result from a variety of processes, such as job analysis, concept hierarchy analysis, and needs assessment procedures. How we arrive at the list of topics or tasks to be included in our system depends on the instructional context, the sociocultural context, the learners being instructed, the management context, and the goal orientation of the educational or training system.

Describing Tasks

Task description is the process of elaborating the tasks, goals, or objectives identified in the inventory. Task descriptions may include listing (a) the tasks included in performing a job, (b) the steps in performing a task, or (c) enabling objectives for a terminal objective. The procedures for performing the task description function depend upon the nature of the information provided in the inventory. Task description always involves an elaboration of the tasks/goals stated in the inventory to a greater degree of specificity or detail. The emphasis here is thoroughness—ensuring that important instructional components are not excluded.

Selecting Tasks

Some instructional development models, especially those in the military, include a separate procedure for selecting from the task inventory those tasks for which training *should* be provided. Since it is impossible to train every person on every task to a level of proficiency that might be required by the job, developers often must select certain tasks for training. According to Tracey, Flynn, and Legere (1970), tasks that are feasible and appropriate for on-the-job, school, and follow-up training should be selected. This selection process may also result from a consideration of various system constraints, such as available time and resources (Davis, Alexander, & Yelon, 1974). In order to select tasks for training, developers need to rank or assign priorities to their training objectives. Task selection is also performed to avoid instructing or training students on material they already know. Thus, those tasks that have already been acquired are eliminated from the list of training objectives. While a task description elaborates the task into its component parts, task selection asks which of these tasks or components are entry level or prerequisite and which tasks are feasible to train. The result of this operation is the final list of training objectives. In many design models, selection is an implicit function, not one that is performed systematically.

Sequencing Tasks and Task Components

The task sequence is often implied by the nature of the tasks in the inventory or the components in the task description. However, the task sequence is more than simply a description of the sequence in which the task is performed. It indicates the sequence in which the instruction occurs. The sequence for performing the task implies an appropriate instructional sequence. For example, the training of employees to perform certain jobs implies a temporal sequence of tasks that models the job. This may not always be the most efficient sequence. Instructional sequencing may also be determined by the content/task analysis process or by the design model being used. For instance, elaboration theory (Reigeluth & Stein, 1983) prescribes a specific top-down, general-to-specific conceptual sequence for presenting material, where learning hierarchy analysis suggests a bottom-up, simple-to-complex sequence. According to taxonomies of learning, different content and different tasks suggest different sequences of instruction. So, sequencing varies according to the theory or model on which it is based.

Analyzing Task and Content Levels

Analyzing task and content levels is the function in the task analysis process in which the mental or behavioral performance required to acquire the task or knowledge is described. That is, designers describe the type of mental behavior, physical performance, or affective response required by the task. This usually takes the form of classifying the task statement according to various learning taxonomies.

Table 1 compares a number of these taxonomies, which describe learning in terms of hierarchies of content. Beginning with the lowest level or most fundamental forms of behavior (reflexes), they describe increasingly more complex mental responses or behavior (evaluation, problem solving, or strategies). The purpose of classifying tasks varies with different models. Normally, however, taxonomic classification of objectives and test items ensures consistency between the goals, the test items, and the instructional procedures. Exact instructional procedures for sequences are implied by some models and hierarchies, such as the component display theory (Merrill, 1983).

Objectives

Another component of the task analysis process that could arguably be included in the list of functions is the instructional or behavioral objectives. They are the most common component

Table 1

Comparison of Taxonomies of Learning

Bloom, 1956	Gagne, 1966, 1977	Leith, 1970	Merrill, 1983*	Mager & Breach, 1967
		Stimulus discrimination		
Knowledge	Information	Response learning	Facts	Memorization
		Response integration		
			Procedures	Procedural
Comprehension	Concrete concepts	Learning set formation		Comprehension
	Defined concepts	Concept learning	Concepts	
Application			Rules	
Analysis	Principles		Principles	
		Hypothetico-deductive inference		
		Learning schemata		
Synthesis	Problem solving			
	Cognitive strategies			
Evaluation				
Affective domain	Attitudes			Attitude development
Psychomotor domain	Motor skills			

*At task level: remember, use and find.

of all instructional development models (Andrews & Goodson, 1980). However, objectives are not a *process*. Rather, objectives are a *product*, resulting from task analysis or some other process. Objectives represent specific statements of the tasks being analyzed. Sometimes, objectives are an input to the task analysis process. That is, objectives are often determined by some process (needs assessment, curriculum guide, fiat) prior to the instructional developer being consulted. So the developer begins by inventorying the tasks limited by the objectives. More commonly, however, a list of objectives and enabling objectives are the product of the task analysis process. They are an essential tool of all of the task analysis functions—inventory, description, selection, sequencing, and analysis—but do not constitute a separate function in the process. While they are essential to the process, for our purposes, they are not part of it.

Needs Assessment

The distinction between task analysis and needs assessment is especially ambiguous, since they are complementary, contributory, and often overlapping processes. Needs assessment, like task analysis, is a process. It is a process that entails three or more functions depending upon definition. It is a formal process for determining the present capability of prospective learners, the desired outcomes, and the discrepancies between the two (Kaufman, 1972). It also frequently entails the ranking of those discrepancies in order of priority. In many respects, needs assessment mirrors task analysis. The sequence is often similar, and there is a variety of procedures available for performing needs assessment functions, some of which are often used to conduct task analysis functions. Yet, when it is performed, needs assessment nearly always precedes task analysis, so that it is usually contributory to task analysis. Needs assessment frequently comprises the task inventory and, with less frequency, the task selection functions of the task analysis process. Therefore, they overlap, and complement each other. However, task analysis is a larger process that does not always depend on needs assessment.

Functions Included in the Task Analysis Process

Task analysis, as performed in various instructional development models, may include some or all of the previously described functions. The task analysis process varies, so the procedures used during the task analysis process may include only one or all of these functions. However, all task analysis procedures performed using various design models can be described by one or more of these functions. That is, these functions, as represented by most task analysis procedures, are usually distinct enough to be identified. Some procedures may perform two or more functions simultaneously. There is no universal temporal sequence in which these phases are performed. As mentioned earlier, Romiszowski (1981) recommends a top-down sequence of inventory, sequencing, analysis, and description. Most designers perform the inventory first, followed by a description. The analysis frequently precedes the sequencing. The functions and procedures used by the developer depend to a large extent on a group of variables to be described next.

Task description always involves an elaboration of the tasks/goals stated in the inventory to a greater degree of specificity or detail.

Task Analysis Variables

The variability in the procedures used to accomplish the task analysis functions results from: (a) the diversity of tasks being analyzed (from psychomotor tasks to complex problem-solving tasks); (b) the instructional situation (from assembly line to experimental laboratory); (c) the characteristics of the learners; (d) the designer's experience and training, and other project constraints, and (e) the instructional development model being applied. The problem is to determine which task analysis procedures are appropriate for accomplishing the task analysis functions. In order to do that, we need to identify the variables that affect the task analysis process and the different functions performed as part of it. These variables can then be used along with the functions as a method for determining the appropriate procedures to be used. A quasi-algorithm is needed for selecting from among available task analysis procedures. In order to do this, we need easily classifiable variables. Some important variables affecting the task analysis process which also lend themselves to classification, are described below.

Micro-Macro

Task analysis procedures are used in different levels of instructional planning. Micro-level procedures are those that pertain to a relatively small portion of instruction, usually an individual objective, a single idea or a single task. Procedures like Component Display Theory (Merrill, 1983) describe how to classify, test, and present instruction for an individual objective. Many traditional behaviorally oriented task analysis procedures, such as behavioral analysis (Mechner, 1967), mathetics (Gilbert, 1961), and learning contingency analysis (Gropper, 1974), analyze each objective for the discriminations, generalizations and chains of behavior required to accomplish it. Even more contemporary task analysis procedures, such as information processing analysis (Merrill, 1978; 1980), analyze individual performances for their information processing requirements. At the micro-level, it is sometimes difficult to see how a single objective or task fits into the entire course. Micro-level analysis is important for determining task requirements and instructional procedures. However, when sequencing tasks, it is important to analyze the tasks from a macro-level to see how the task requirements fit together.

Macro-analysis usually implies unit or course level analysis. Knowing how to integrate and summarize more than one idea, task, or objective and synthesize them into a meaningful sequence is also an important task analysis function. Procedures such as elaboration theory (Reigeluth & Stein, 1983) provide specific guidelines based upon cognitive instructional theory for organizing and sequencing the components of a course. Concept hierarchy analysis (Tiemann & Markle, 1983) is a process for analyzing the conceptual components of subject matter. The most prominent task analysis procedure, learning hierarchy analysis (Gagne & Briggs, 1979) also operates at a macro-level, although not always at a course level. Rather, it is used to identify and sequence the prerequisite skills or performances that lead to course goals. In order to design instruction successful, it is necessary to develop this larger picture on how content is organized. The procedures used to do that are different from micro-level procedures.

The problem is to determine which task analysis procedures are appropriate for accomplishing each task analysis function.

Top-Down Bottom-Up

Task analysis procedures vary also in terms of their overall approach to analyzing tasks. Those procedures that are more concerned with content or concept analysis take a *top-down approach*. That is, they begin at the most general or abstract level of content or with the most general task description and proceed to break it down into its component concepts or tasks. Top-down analysis then is an elaborative process, seeking more detail and specificity. Learning hierarchy analysis (Gagne & Briggs, 1979), for instance, begins with a generic task and analyzes it for its prerequisite tasks, and those for their prerequisites and so on. Information processing analysis (Merrill, 1978; 1980) starts with a task and looks on a micro-level at the specific mental process that produce that performance. Top-down task analysis procedures proceed from the general to the specific in a hypothetico-deductive fashion.

Bottom-up task analysis procedures, on the other hand, start at the specific level and build up an instructional sequence. They proceed from the single task or steps in a task and proceed to construct a task sequence from it. This type of analysis is most common in job task analysis (Mager & Beach, 1967) where a designer starts by observing a sequence of steps involved in performing a task. The critical incident technique (Flannigan, 1954; Zemke, 1981) is also a bottom-up process, where analysis begins with describing the critical incidents in job performance. Bottom-up analysis procedures are specific-to-general, inductive types of analysis processes. In most industrial settings, they are helpful in analyzing job task requirements.

Job Task Analysis vs. Learning Task Analysis

An important distinction to task analysis is the source of the task and the orientation of the agency developing the tasks. Is the task being analyzed a job task or a learning goal or objective? That is, is it a job task analysis or learning task analysis? Is the agency developing training or educational sequences? Job task analysis occurs more commonly in business and industry, while learning task analysis is practiced more commonly in educational institutions.

Job task analysis is normally undertaken to solve a performance problem. Learning task analysis, on the other hand, is undertaken to develop a curriculum. The reasons for conducting task analysis will affect the nature of the process. While the curriculum resulting from a learning task analysis may prepare learners to perform the same jobs or roles for which job task analysis is used to develop training, the goal-orientation of the agencies conducting the analysis is different. Developers who design training sequences seek to develop mastery of specific tasks, whereas developers who design learning sequences usually are more concerned with mastery of subject matter knowledge. These orientations are reflected in processes normally referred to as job task analysis and learning task analysis. Educators foster knowledge acquisition; this approach is proactive. Trainers, on the other hand are more reactive, engaged in an ad hoc attempt to rectify problems. Educators design pre-service instruction, whereas the trainer/developer tend to design in-service training. The focus, orientation, and purpose of these two entities are usually disparate.

This difference in orientation is also reflected in the nature of the knowledge and tasks being analyzed. The job trainer is more concerned with procedural knowledge—how to do something or perform some task. The educator is more concerned with conceptual knowledge—the ideas, concepts and principles and their interrelationships that constitute a field of study. The former usually results in near transfer of training, while the conceptual approach more often produces far transfer (Clark & Voogel, 1985). Job training is not as concerned with getting trainees to apply or transfer their skills to similar problems in different settings. Since educators do not know the specific settings into which their students will go, they must be more concerned with far transfer, that is, the ability of their students to apply knowledge in a broad range of settings. Trainers, therefore, tend to use more behavioral training methods, while educators stress cognitive processes. Behavioral methods promote near transfer; cognitive methods promote far transfer (Clark & Voogel, 1985). While industry and the military rely more on training, there are many educators in their ranks, just as a lot of training is conducted in traditional educational institutions.

These three variables are somewhat global classifications of task analysis procedures. However, when combined with the task analysis functions, they can be used to make recommendations for the task analysis procedures that should be employed. In the next section, these variables are combined to form a quasi-algorithm for making general recommendations regarding selection of appropriate task analysis procedures.

Selecting Task Analysis Procedures

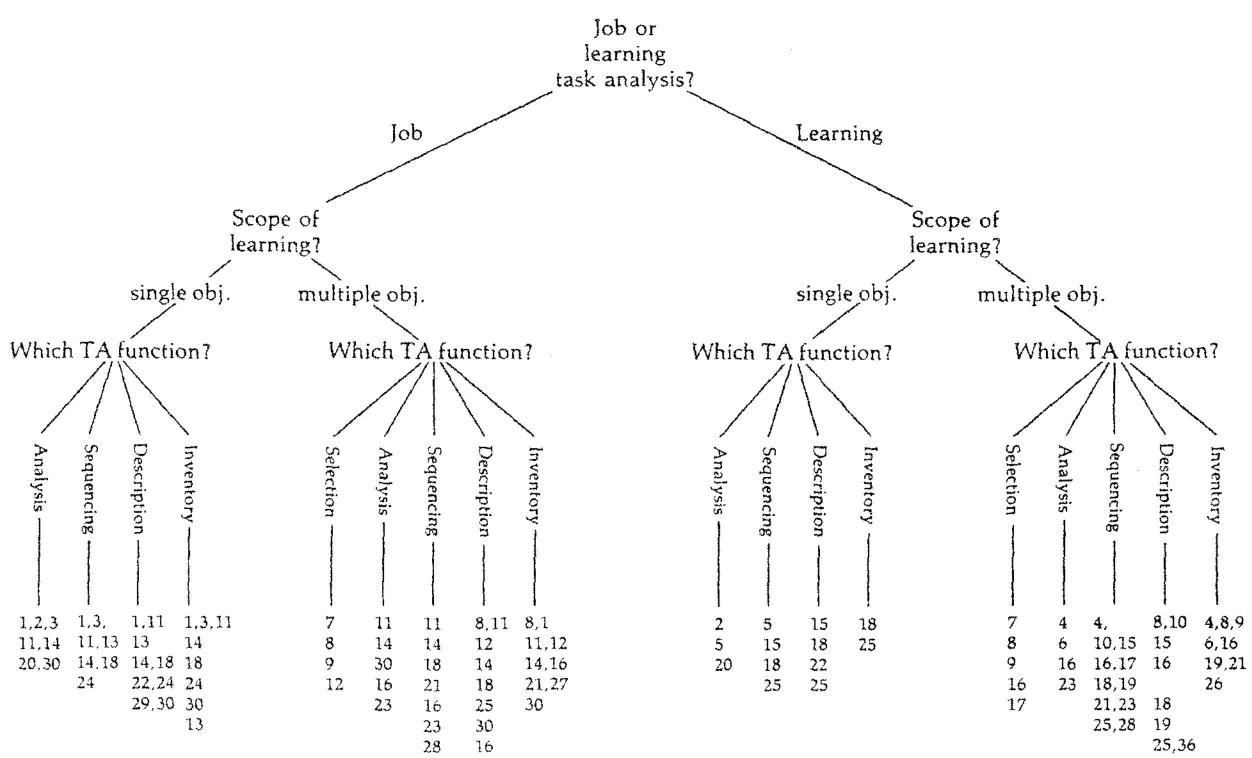
So far, we have described the ambiguity in the task analysis process and provided a scheme for describing and classifying task analysis procedures. The problem of which procedure to use to accomplish each task analysis function remains. We know that the ability to make informed judgements depends on experience. Experienced developers recommend task analysis procedures for use in different situations based upon their better developed "scripts" for the instructional development process. The purpose of this article then, is to use our organizational scheme to make suggestions about which task analysis procedures may be used for each function. Based upon his review of three task analysis technologies, Foshay (1983) made some informed recommendations about which task analysis procedures would be appropriate under different conditions. For instance, he recommended learning hierarchy analysis for macro-level sequencing, concept hierarchy analysis for discriminating among concepts, and so on. However, his review considered only three of the many task analysis procedures available to developers.

In Figure 1, we present a quasi-algorithm for selecting alternative task analysis methodologies. It is our belief that selecting from the many available procedures is best done through a sequence of decisions. The divisions in this algorithm are based upon the classifications of task analysis procedures previously discussed: (a) functions (inventorying, describing, selecting, sequencing, and analyzing) and (b) variables (micro-macro, top-down/bottom-up, and job vs. learning task analysis). In order to use the algorithm, first decide whether you are conducting a job analysis or an instructional analysis. That is, are you designing training for a specific job or are you developing a general unit of instruction? Next, consider the scope of learning. Are you developing instruction for a single task or objective or a set of course objectives? Are you operating at a macro-level or micro-level? Finally, decide which of the task analysis functions you are performing—inventory, description, selection, sequencing, or analysis. As you make this sequence of decisions and follow the appropriate paths, you are led to one or more numbers, which are keyed to the task analysis procedures listed and annotated in the Appendix. The numbered procedures shown at the bottom of each decision path in Figure 1 are the appropriate procedures which may be used to accomplish the task analysis function in the setting implied by the decisions. The choice of which procedure to use depends upon the experience and/or preferences of the designer or some organizational decision by a design team.

Conclusion

It is not our intention to offer a definitive prescription about which specific task analysis procedure should be used for every function in every setting. The knowledge about the task analysis process is too uncertain for us to make specific recommendations about which procedures to use to solve all design problems. Rather, we have tried to impose some organization on the task analysis process. In doing so, we hope to provide some guidance to the

Figure 1. Algorithm for Selecting Task Analysis Methodologies.



*The suggestions shown here are based on the normal, intended purposes for each method. So, they are not exhaustive. It is possible to innovatively apply each method to a variety of functions.

beginning developer in selecting the procedures that *could* be used to accomplish the various task analysis functions in different settings. Once you have used the algorithm to narrow your choices to a given category, you must familiarize yourself with the alternative procedures in order to make the final selection of task analysis procedures to be used.

References

Andrews, D. H., & Goodson, L. A. (1980). A comparative analysis of models of instructional design. *Journal of Instructional Development, 3* (4), 2-16.

Bloom, B. S., Krathwhol, D. R., & Masia, B. B. (1956). *Taxonomy of educational objectives: The classification of educational goals. Handbook I: Cognitive domain.* New York: David McKay.

Buzan, T. (1974). *Use both sides of your brain.* New York: E. P. Dutton.

Clark, R. E., & Voogel, A. (1985). Transfer of training principles for instructional design. *Educational Communications and Technology Journal, 33* (2), 113-123.

Dalkey, N. C., & Helmer, O. (1963). An experimental application of the Delphi Method to the use of experts. *Management Science, 9,* 458-467.

Davies, I. K. (1976). *Objectives in curriculum design.* New York: McGraw Hill.

Davis, R. H., Alexander, L. T., & Yelon, S. L. (1974). *Learning system design: An approach to the improvement of instruction.* New York: McGraw-Hill.

Design of courses of instruction (Report No. MCOP11510 23A) (1972). Washington, D. C.: U. S. Marine Corps.

Diekhoff, G. M., & Diekhoff, K. B. (1982). Cognitive maps as a tool in communicating structural knowledge. *Educational Technology, 22* (4), 28-30.

Evans, J. L., Glaser, R., & Homme, L. E. (1962). The RULEG system for the construction of programmed verbal learning sequences. *Journal of Educational Research, 55,* 513-518.

Fields, A. (1982). Getting started: Pattern notes and perspectives. In D. H. Jonassen (Ed.). *The technology of text: Principles for structuring, designing, and displaying text,* (Vol. 1). Englewood Cliffs, NJ: Educational Technology Publications.

Fine, S. A., & Wiley, W. W. (1971). *An introduction to functional job analysis.* Kalamazoo, MI: Upjohn.

Flannigan, J. C. (1954). The critical incident technique. *Psychological Bulletin, 51,* 327-358.

Foshay, W. R. (1983). Alternative methods of task analysis: A comparison of three methods. *Journal of Instructional Development, 6* (4), 2-9.

Fussell, J. B., Powers, G. J., & Bennett, R. (1974). Fault trees: A state of the art discussion. *IEEE Transactions on Reliability, R-23,* 5-55.

Gagne, R. M. (1965). *The conditions of learning* (1st ed.). New York: Holt, Rinehart & Winston.

Gagne, R. M. (1974). Task analysis—Its relation to content analysis. *Educational Psychology, 11* (1), 11-18.

Gagne, R. M. (1975). *Essentials of learning for instruction.* Hinsdale, IL: Dryden Press.

Gagne, R. M. (1977). *The conditions of learning* (3rd ed.). New York: Holt, Rinehart, & Winston.

Gagne, R. M. (1985). *The conditions of learning* (4th ed.). New York: Holt, Rinehart & Winston.

Gagne, R. M., & Briggs, L. J. (1979). *Principles of instructional design* (2nd ed.). New York: Holt, Rinehart & Winston.

Gentry, C. (1985, January). *Needs analysis: Rationale and technique.* Paper presented at the annual meeting of the Association for Educational Communications and Technology, New Orleans.

Gilbert, T. F. (1961). Mathetics: The technology of education (Special issues). *Journal of Mathetics, (1* and *2).*

Gilbert, T. F. (1982a). A question of performance—Part I: The PROBE model. *Training and Development Journal, 36,* 21-30.

Gilbert, T. F. (1982b). A question of performance—Part II: Applying the PROBE model. *Training and Development Journal, 36,* 85-89.

Gropper, G. L. (1974). *Instructional strategies.* Englewood Cliffs, NJ: Educational Technology Publications.

Harless, J. H. (1980). Task analysis—A clarification of the term. *NSPI Journal, 19* (2), 4-5.

Harrow, A. J. (1982). *A taxonomy of psychomotor domain.* New York: David McKay.

Hershbach, D. R. (1976). Deriving instructional content through task analysis. *Journal of Industrial Teacher Education, 13* (3), 63-73.

Hoffman, C. K. & Medsker, K. L. (1983). Instructional analysis: The missing link between task analysis and objectives. *Journal of Instructional Development, 6* (4), 17-23.

Job task analysis manual (1973). San Diego, CA: Naval Training Center—Service Schools Command, Industrialized Learning Development Group.

Jonassen, D. H. (1984). Developing a learning strategy using pattern notes: A new technology. *Programmed Learning and Educational Technology, 21* (3), 163-175.

Jonassen, D. H. (in press). Verifying a method for assessing cognitive structure using pattern notes. *Journal of Research and Development in Education.*

Kaufman, R. (1972). *Educational system planning.* Englewood Cliffs, NJ: Prentice Hall.

Kennedy, P., Esquire, T., & Novak, J. (1983). A functional analysis of task analysis procedures for instructional design. *Journal of Instructional Development, 6* (4), 10-16.

Krathwohl, D. L., Bloom, B. S., & Masia, B. B. (1964). *Taxonomy of educational objectives, the classification of educational goals. Handbook II: The affective domain.* New York: David McKay.

Leith, G.O.M. (1970). The acquisition of knowledge and mental development of students. *(British) Journal of Educational Technology, 1,* 116-128.

Mager, R. F., & Beach, K. M. (1967). *Developing vocational instruction.* Belmont, CA: Fearon.

Martin, M. C., & Brodt, D. E. (1973). Task analysis for training and curriculum design. *Improving Human Performance Quarterly, 2,* 113-120.

McCormick, E. J. (1979). *Job analysis: Methods and applications.* New York: American Management Association.

McDermott, F. M. (1982). Try brainstorming—a quick route to job analysis. *Training/HRD, 19* (3), 38-40.

Mechner, F. (1967). Behavioral analysis and instructional sequencing. In P. C. Lange (Ed.), *Programmed instruction.* Yearbook of the National Society for the Study of Education. Chicago: Rand McNally.

Merrill, M. D. (1975). Learner control: Beyond aptitude—treatment interactions. *A Communication Review, 23,* 217-226.

Merrill, M. D. (1983). Component display theory. In C. M. Reigeluth (Ed.), *Instructional design theories and models: An overview of their current status.* Hillsdale, NJ: Erlbaum.

Merrill, P. F. (1978). Hierarchical and information processing task analysis: A comparison. *Journal of Instructional Development, 1,* 35-40.

Merrill, P. F. (1980). Analysis of a procedural task, *NSPI Journal, 19* (2), 11-15, 26.

Miller, R. B. (1962). Task description and analysis. In R. M. Gagne & A. W. Melton (Eds.), *Psychological principles in system development.* New York: Holt, Rinehart and Winston.

Reigeluth, C. M., Merrill, M. D., & Bunderson, C. V. (1978). The structure of subject matter content and its instructional design implications. *Instructional Science, 7* (2), 107-126.

Reigeluth, C. M., Merrill, M. D., Branson, R. K., Begland, R., & Tarr, R. (1980). *Extended task analysis procedure (ETAP): User's manual.* Final Report, Army Training Development Institute. (NTIS No. AD-A098351)

Reigeluth, C. M., & Rogers, C. A. (1980). The elaboration theory of instruction: Prescriptions for task analysis and design. *NSPI Journal, 19* (1), 16-26.

Reigeluth, C. M., & Stein, F. S. (1983). The elaboration theory of instruction. In C. M. Reigeluth (Ed.), *Instructional design theories and models: An overview of their current status.* Hillsdale, NJ: Erlbaum.

Resnick, L. B. (1976). Task analysis in instructional design: Some cases from mathematics. In D. Klahr (Ed.), *Cognition and instruction.* Hillsdale, JN: Erlbaum.

Resnick, L. B., & Ford, W. W. (1982). The analysis of tasks for instruction: An information processing approach. In T. A. Brigham & A. C. Catania (Eds.), *Handbook of applied behavior analysis: Social and instructional processes.* New York: Irvington.

Romiszowski, A. J. (1981). *Designing instructional systems.* London: Kogan Page.

Stone, P. J., Dunphy, D. C., Smith, M. J., & Ogilvie, D. M. (1966). *The general inquiries: A computer approach to content analysis.* Cambridge, MA: MIT Press.

Thomas, C. A., Davies, I. K., Openshaw, D., & Bird, J. B. (1963). *Programmed learning in perspective: A guide to program writing.* Chicago: Educational Methods.

Tieman, P., & Markle, S. (1983). *Analyzing instructional context: A guide to instruction and evaluation* (2nd ed.). Champaign, IL: Stipes.

Tracey, W. R., Flynn, E. B., & Legere, C. L. J. (1970). *The development of instructional systems.* Fort Devens, MA: U. S. Army Security Agency Training Center.

Zemke, R. (1981). The critical incident method of analysis. In R. Zemke, L. Standke & P. Jones (Eds.), *Designing and delivering cost effective training and measuring the results.* Minneapolis, MN: Lakewood.

Appendix

Task Analysis Methodologies

1. Behavioral Analysis. Like many other task analysis procedures, behavioral analysis (Mechner, 1967) grew out of programmed learning. In an attempt to develop systematic methods for sequencing frames of programs, Mechner suggested analyzing the components of each objective. Like Gilbert (1961) and Gropper (1974), he classified these components as discrimination, generalizations, or chains. He developed a set of rules for sequencing chains (procedures) and concepts, such as "never teach a discrimination without simultaneously teaching a generalization" (p. 94). The instructional developer can perform a behavioral analysis by merging the types of questions students might ask about discriminations, generalizations, and chains, such as "What are the steps at arriving at this conclusion?", "Where is all this leading?", or "What are some examples of concepts?" To the extent that we feel comfortable in generalizing programmed learning procedures, behavioral analysis provides a useful means for micro-level task analysis and sequencing of instruction.

2. Bloom's Taxonomy. Bloom and his colleagues (Bloom, Krathwohl, & Masia, 1956; Krathwohl, Bloom & Masia, 1964) spent several years developing a taxonomic classification of cognitive and affective behaviors for purposes of test design. A taxonomy of psychomotor domain was added later (Harrow, 1972). These taxonomies later became the primary means for analyzing learning tasks. They describe in detail increasingly complex forms of cognitive behaviors (from knowledge to evaluation), affective behaviors (from receiving to articulation of a value concept), and psychomotor behaviors (from imitation to naturalization). These remain the most detailed descriptions of learning behaviors, still popular with many educators (see Table 1).

3. Brainstorming. Brainstorming provides a quick route to job analysis (McDermott, 1982). The developer assembles skilled job performers in order to determine the model job performance. All steps and functions are posted on index cards on a large, clear wall. Using different color cards, all contingencies are posted for each step. Then the developer tries to get consensus on the most realistic alternatives to each of the listed contingencies. Finally, the knowledge and skill requirements for each step are stated. This brainstorming procedure is a quick and easy method for analyzing jobs. Its strength lies in the elaboration of contingent behaviors necessary for performing the job.

4. Cognitive Mapping. Understanding concepts is necessary but insufficient for understanding content. Learners must also understand the structural relationships between related concepts. So if we use content or concept analysis procedures for identifying concepts, we will need a method to derive the type and degree of relatedness among those concepts. Cognitive mapping provides a tool for this (Diekhoff & Diekhoff, 1982). Once the key concepts are selected, designers or subject matter experts should form all possible pairs of those concepts and rate each pair for degree of relatedness using a 1-9 scale. The relatedness matrix is treated as an intercorrelation matrix and analyzed using principal components analysis or multi-dimensional scaling. The output of the analysis is a map that spatially relates the inter-concept distances. This process could aid both the sequencing and analysis phases. Sequencing is aided because the clusters that are formed indicate content groups. While not a traditional form of taxonomic analysis, the meaning of concepts is enhanced by knowing relationships among concepts. Further analysis of these relationships adds another dimension of meaning (Jonassen, 1984).

5. Component Display Theory. The component display theory (Merrill, 1983) is a micro-level design strategy for organizing instruction for a single idea or objective in the cognitive domain. The designer begins by classifying each objective to be taught in terms of the nature of the task and the content, a distinction missing from most analysis schemes. An objective can require the learner to remember, use, or find either facts, concepts, procedures, or principles (see Table 1). Component display theory recommends the use of four primary presentation forms (tell or ask generalities or instances) and six types of elaboration (context, prerequisite, mnemonic, mathemagenic help, representation, feedback). It then provides rules that state the required primary presentation forms and elaborators for different types of tasks and content. While component display theory is an instructional design system, much of which is used after task analysis, the task/content matrix is very useful for the analysis phase because of its explicitness.

6. Conceptual Hierarchy Analysis (Tiemann & Markel, 1983; Reigeluth, Merrill & Bunderson, 1978). The sequencing of instruction, according to concept hierarchy analysis, is implied by the structure of the content. Various content structures (description, comparison/contrast, temporal sequence, explanation, definition/examples, problem/solution, cause/effect) may suggest different sequences for different tasks. Concept hierarchy analysis is a macro-level task analysis procedure for identifying, organizing and arranging instructional content in the absence of a specific procedure. It requires identifying and analyzing the network of concepts used in any content area.

7. Criteria for Task Selection. Most of the military task analysis processes include an explicit procedure for selecting from among tasks or objectives those in which training should be provided (*Design of courses of instruction*, 1973; *Job task analysis manual*, 1973; Tracy, Flynn, and Legre, 1970). The criteria for determining feasibility and appropriateness include: universality (transferability), difficulty of acquisition, cruciality to the mission, frequency of performance, practicability, achievability by trainees, quality of skill, deficiencies resulting from training, retainability, and need for follow-up training. With limited training resources, a broad range of skills to cover, and a large number of trainees, the military is obviously pressed to develop comprehensive training. These task selection criteria help to rank the importance of each task in order to provide training for the most important tasks first. While these criteria are seldom applied to educational (learning) problems, they could be.

8. Critical Incident Technique. Determining the tasks to be included in instruction is often accomplished by using critical incident analysis (Flannigan, 1954; Zemke, 1981). In this technique, experts identify the critical job incidents and their products. Incidents are edited for redundance, grouped into similar tasks, and then classified as positive or negative incidents. The incidents are summarized and then validated by the experts for completeness. This is a useful means for obtaining a list of relevant, real-world tasks to be included in instruction. It is a job-related technique, however, and is most useful for converting job descriptions into instructional inventories.

9. Delphi Technique. In selecting the tasks/content to be taught, it is often necessary to place the inventory in priority order. This often requires the informed judgements of subject matter experts. One of the most popular techniques for generating that data is the

Delphi technique (Dalkey & Helmer, 1963), in which sets of comments/beliefs/questions are submitted to an anonymous group of subject matter experts for their judgements. Their responses are analyzed and summarized, and then become the questions for the next round of judgements. This iterative judgment-feedback cycle is continued until the panel reaches consensus. The result represents the convergent thinking of a group of experts. It can be a tedious process, but it is one of the most systematic for collecting judgements.

10. Elaboration Theory. The elaboration theory (Reigeluth & Rogers 1980; Reigeluth & Stein 1983) provides a simple-to-complex approach to organizing instruction in which concepts, procedures, or principles are iteratively detailed and epitomized. It is a macro-level strategy for organizing multiple objectives. For each single objective, component display theory is used to organize instruction. That is, instruction starts at a general level with an epitome; (i.e., the organizing of content ideas). These general ideas are then elaborated in progressively more detailed steps. Each level of elaboration has its own epitome (overview), which indicates the content structure of that elaboration, a summarizer (e.g., statement, example, or self-test), and a synthesizer to integrate that level of elaboration to all higher level elaborations. In addition, elaboration theory employs strategy components, such as analogies, cognitive strategies, and learner control. Elaboration theory views task analysis as a form of content analysis; from that point of view it supports the task inventory, description, and sequencing functions. The analysis steps include selecting the operations to be taught, deciding which to teach first, sequencing the remaining operations, creating the epitomes, and designing instruction on each operation (Reigeluth & Rogers, 1980). Performed in the context of elaboration theory, these represent a comprehensive and systematic top-down approach to learning analysis that is seldom ever used to organize job-related training.

11. Extended Task Analysis Procedure. The extended task analysis procedure (ETAP) (Reigeluth, Merrill, Branson, Begland, & Tarr, 1980) is a 12-step process for analyzing procedural tasks that combines hierarchical and information processing analysis procedures. It was developed for the military specifically to support job training. The three phases of the process include process analysis (identifying each step using information processing analysis), sub-step analysis (identifying the sub-steps for each step), and knowledge analysis (identifying the knowledge required to perform the task). The result is a multi-dimensional representation of the learning task including a flowchart, a list of sub-steps, and a list of component facts and principles. What is unique to ETAP is the factor-transfer and principle-transfer analysis. In complex transfer tasks that include a large number of conditions or factors, ETAP identifies all the factors and creates decision rules and more general common rules for dealing with those factors in a transfer situation. Where those factors cannot be identified easily, ETAP identifies and sequences into instruction the necessary principles for properly executing the transfer task. Attention to this transfer of training is often absent in instructional design models, especially in the task analysis process.

12. Fault Tree Analysis. Another method for selecting the tasks to be taught focuses on avoiding errors or faults. Fault tree analysis (Fussell, Powers & Bennett, 1974) predicts undesired events that may affect the operation of a system and provides the basis for redesigning it to prevent those occurrences. It can be used to select those tasks necessary for preventing undesired events. The result of such an application of fault tree analysis is a priority list of training needs. Working backward from a statement of an undesired event (previously identified), fault tree analysis represents all antecedent conditions that could have caused the event. The same process is repeated for each of those events, with each causal condition represented by an AND or OR logic gate. This process produces a tree of causal events, which shows each of the critical paths that produce the undesired event and the probability of the occurrence of each. Working with this information the designer could select those paths with the highest probability of occurrence as the most important training needs. This is a technical procedure that also requires a thorough knowledge of the operation system by the developer in order for it to be successful (Gentry, 1985).

13. Functional Job Analysis. Functional job analysis conceptually defines worker activity and defines methods for measuring worker output (Fine & Wiley, 1971). All jobs require workers to relate to data, people, and things (machines). Each job can be defined in terms of the workers' interactions with these three elements. Those interactions are actually limited. That is, there are only a few ways the workers can interact with certain types of machines. The job functions related to these three elements are sequential and hierarchical, proceeding from simple to complex. In that sense, it is much like learning hierarchy analysis, which specifies all of the prerequisite tasks to each goal. So analysis of any job task describes how the worker relates to data, people, and things as well as the relative amount of involvement he/she has with each element. This comprehensive analysis of job tasks has been adopted by several private and governmental organizations as their job analysis procedure.

14. Job Task Analysis (Mager and Beach, 1967). In the context of developing vocational instruction, the task analysis procedures focus on job description—what a worker does under the conditions that the job is normally performed, rather than what you would like him/her to do. The procedure requires the designer to list all of the tasks in a job and the steps included in each task; i.e., what a person does when performing the step, the type of performance involved (see Table 1), and the expected difficulty in learning it. From the task analysis, the designer derives course objectives after first determining what the learners already know. Course objectives, then, describe those things that learners should be able to do at the end of the course. Except for the determination of the type of performance required by each step, this is a vocational, behavioral analysis technique that focuses on the inventory function.

15. Information Processing Analysis (Merrill, 1978; 1980; Resnick, 1976; Resnick & Ford, 1982). Similar to learning hierarchy analysis, information processing analysis describes the sequence of cognitive operations required for solving a class of problems. Such analysis usually represents the information processing sequence in algorithmic form. The goal of such analysis is to model the covert mental operations of a learner while performing a task, rather than modelling the overt behavior exhibited by the learner. While it is normally applied to problem solving, information processing analysis may be used to describe other tasks. Such analysis must be generic so that it may be applied to a range of problems (tasks). It may imply a forward or a backward sequence of development, depending upon the problem solving technique employed. (See also Path Analysis, 25).

16. Instructional Analysis. Instructional analysis is a comprehensive set of task analysis procedures intended as a critical link between task analysis and writing instructional objectives (Hoffman & Medsker, 1983). By analyzing the component skills, instructional analysis seeks to identify "New learning," excluding those skills already known from a list of "instructional" objectives. So, after identifying and sequencing component skills and eliminating extraneous ones, the instructional analyst identifies the type of learning required by the remaining skills using a hybrid taxonomy. This taxonomy includes complex procedures which are pre-defined, interrelated sequences of operations that can be considered a unit. So, starting with a task analysis, the instructional analyst analyzes the type of learning and conducts a traditional hierarchical analysis, a procedure analysis, or a combination analysis which combines the complex procedures. After identifying support skills not integral to the task, a learning map that combines all of the previous analyses is constructed. Instructional analysis is a super-procedure that adds to task analysis. It represents one of the most comprehensive task analysis processes available.

17. Learner Control of Instruction (Merrill, 1975). Learner control describes an instructional strategy rather than a procedure for designing instruction. Essentially, it argues for allowing the learner some degree of self-determination of the content and strategies of instruction (Merrill, 1983). The content may consist of the objectives, lesson, or module selected by the learner. It has the most significant implications for task analysis in the sequencing and selection functions. Giving students the opportunity to select what they will learn as well as the order in which they will complete instruction can preclude some of the sequencing operations normally performed by the designer. To responsibly select instructional content requires some metacognitive skills, which many learners do not possess. Because of this, the research findings related to learner control have been mixed, at best.

18. Learning Contingency Analysis. A task inventory or description provides a set of tasks, or steps in a task, and the ordering of these. Usually performance of one task/step is contingent on another, which is contingent on a prior skill. Since these contingencies have implications for instructional sequences, designers can develop a corresponding progression of steps to be taught. The progression or sequence is dependent on the relationships among tasks/steps. A learning contingency may be necessary, facilitative, or non-existent depending upon four types of relationships: superordinate/subordinate, coordinate input/output, shared elements, or no relationship (Gropper, 1974). The sequence in which behavioral components should be learned in turn depends upon the nature of the relationship. For instance, Gropper (1974) suggests that an output that becomes an input for another performance should be taught first. This type of task analysis describes the behavioral components of an objective, rather than the traditional taxonomies that are used to describe the terminal performance depicted by the objective.

19. Learning Hierarchy Analysis (Gagne, 1965, 1974, 1975, 1977, 1985; Gagne & Briggs, 1979). Learning hierarchy analysis has become so universal that many equate it with task analysis. Based on his own taxonomies of learning (Gagne, 1965, 1977, 1985), Gagne has described a method for developing a hierarchy of learning skills (see Table 1) for organizing learning tasks. While it could be used to organize instruction for job tasks, it is commonly associated with learning analysis. This is a backward chaining technique for elaborating the prerequisite skills for accomplishing an instructional objective. Learning hierarchy analysis has evolved from a behavioral analysis method for describing the structure of a task and the essential prerequisite skills that comprise that task. For any objective, learning hierarchy analysis describes the prerequisite concepts, principles and strategies necessary for acquiring the skill implied by the terminal objective. The optimal sequence of instruction can be inferred from such learning hierarchies.

20. Learning Taxonomy (Leith, 1970). While structurally similar to Gagne's taxonomy, Leith's (1970) taxonomy (see Table 1) provides specific instructional suggestions in the form of conditions. Leith devoted as much of his hierarchy to associative processes as Gagne did in his earlier work. The primary difference is at the higher end of the taxonomy, where Leith included problem solving and schemata development. Schemata are general networks of ideas and operations. This reference to schemata reflects the shift in the sixties toward a more cognitive orientation in the psychology of learning.

21. Master Design Chart. One means for using objectives to plan curriculum is to develop a master design chart (Davies, 1976). A master design chart is a matrix, with one axis listing content areas and the other listing specific behaviors (objectives). In designing such a chart, the designer first identifies the objectives along the behavioral axis. Second, the content of subject matter is broken down and displayed along the content axis. Third, each cell in the matrix should be evaluated for the emphasis on each type of behavior that should be manifest for each area of content. The resulting matrix reflects the emphasis of the curriculum and could be used to sequence the tasks in a course. It could also be used in a more top-down way at the front end to inventory the tasks to be included in an instructional unit. The master design chart is an alternative method of matrix analysis.

22. Mathetics. Emerging from the programmed instruction movement, mathetics was promoted by Gilbert (1961) as the technology of education, a complete system for task analysis and instructional design. This behavioral approach diagrammatically represented the task sequence that was established by observing and analyzing a master performer. The task analysis classified behavior as consisting of chains, multiple discriminations, and generalizations. Rather than classifying objectives, this taxonomy describes the processes that comprise an objective (Gropper, 1974). Gilbert's concern with the stimulus portion of the S-R association resulted in a specific set of instructional procedures based on the task analysis. These procedures include demonstrating, prompting, or releasing the learner. Gilbert also suggested rules for deciding what content to include and the sequence in which it should be presented. While mathetics has not lived up to his prediction as the technology of education, it represents one of the most comprehensive behavioral task analysis systems available.

23. Matrix Analysis. Like many task analysis procedures, matrix analysis (Evans, Glaser & Homme, 1962; Thomas, Davies, Openshaw, & Bird, 1963) emerged from the programmed instruction literature as a means for sequencing program frames. In designing programs (or other forms of instruction), designers first identify the important concepts and convert those into a set of specific rules. The rules should then be sequenced in some order. In order to adequately communicate knowledge, the interrelationships among rules need to be understood and taught. In order to identify all of the pertinent interrelationships, a matrix is created. The matrix, which shows all possible interrelationships, requires that the designer do a pairwise or cell-by-cell assessment of the relatedness between each possible pair of rules. Each pair is classified as an association (the rules are related and similar) or discrimination (the rules are related but different). The sequence of instruction is reflected in the matrix, so that by observing the matrix, the designer can quickly discern omissions, inverted or misplaced rules or any other sequencing problem. From the matrix, a flow diagram describing the different types of frames is developed, showing the final sequence of instruction. Matrix analysis could be used to help sequence any form of instruction.

24. Methods Analysis. Methods analysis is a micromotion analysis of any job based on detailed motion studies (McCormick, 1979). These often use operation charts that describe in detail the actions of workers at a single location, using standardized symbols to depict each motion of the worker. Micromotion studies analyze videotapes of workers performing jobs in terms of basic motions and develop a simultaneous motion cycle chart that describes the motions of each hand and the body. This type of micro-level analysis is useful for deriving the description phase for psychomotor tasks.

25. Path Analysis (Merrill, 1978, 1980). Path analysis is the second phase of information processing analysis. In conducting a path analysis, the designer identifies the unique paths through an information processing flow chart. This is especially important when a process contains iterative sub-processes. Paths are depicted by listing the numbers of all the operations on a flow chart that the learner executes going from start to stop. Comparing the sequence and inclusiveness of different paths provides a meta-level analysis of the information processing that occurs. This analysis shows the superordinate/subordinate relationships among various paths. That is, some paths may be embedded hierarchically in other paths. Those paths (representing skills) that are subordinate to others are also prerequisite to them, so that learning hierarchy analysis (Gagne, 1965, 1977, 1985) can then be used to analyze the skills. These hierarchical paths are then converted into task sequences for orienting instruction. (See also Information Processing Analysis, No. 15).

26. Pattern Noting. Pattern notes were originally conceived as a notetaking method (Buzan, 1974; Fields, 1982) for summarizing the content of notes in a network map form. To construct a pattern note, you box the key issue or item in the center of a clean sheet of paper. You begin to free associate related topics and write those on lines connected to the box. Sub-issues are written on lines linked to the initial lines. You continue to elaborate the lines until the related topics are complete, and then interconnect any related topics on the map with lines. Pattern notes are excellent organizational and retrieval strategies (Jonassen, 1984) that reflect a person's cognitive structure (Jonassen, in press). They can assist the task analysis process most in terms of the inventory and description functions when the content of instruction is being identified. They are conceptual in nature, so they could support concept hierarchy analysis. Pattern noting, as a measure of cognitive structure, is also a useful measure of prior learning. Pattern noting can depict interrelatedness of prior knowledge, rather than a unidimensional, single score on a pretest. It is similar to, though distinctly different from, concept mapping (No. 4).

27. PROBE Model. The PROBE model (Gilbert, 1982a, 1982b) is a performance analysis procedure that consists of eight sets of questions that analyze the capabilities of workers and the environments in which they work. These individual difference and environmental questions concern the inspiration and instrumentation available to employees as well as the motivational contingencies that result in performance. The questions are used to analyze any performance problem situation in terms of employee skills and motives, knowledge and training, adequate information and feedback, proper tools and responses, and appropriate incentives. The PROBE model is a conceptually sound and practical performance analysis process. It was not designed as a task analysis procedure; it is broader in scope. It could, however, yield useful information to anyone performing a task analysis. The questions related to knowledge and training function as a needs assessment procedure that would supply the basis for task analysis. So, the PROBE model is a useful strategy supporting the task analysis procedure.

28. Syntactic Analysis (Stone, Dunphy, Smith, & Ogilivie, 1966). One of the most difficult parts of task analysis is organizing a large number of tasks that have been inventoried. Syntactic analysis reviews each task statement syntactically, (i.e., looks for statements with similar terms, performing the same syntactic function). For instance, task statements can be analyzed for common direct objects. Those with common direct objects, indicating various performances on the same object, cluster together (Martin & Brodt, 1973). Syntactic analysis can also search for synonyms of objects or other syntactic elements. It is used primarily to order task statements.

29. Task Description (Miller, 1962). A task description specifies the sequence of stimulus-response associations required to complete a task (Miller, 1962). This includes specification of the cues or indicators perceived by the performer, the task activities, and the conditions surrounding each performance required for accomplishing each task. Task analysis further clarifies the behavioral requirements of the task where the designer looks for some behavioral structure in the task. The task description and analysis process, according to Miller (1962) is a molecular process concentrating only on the behavioral aspects of performance.

30. Vocational Task Analysis. Hershbach (1976) proposed a three-step task analysis model that includes a task inventory, a task description, and a task analysis. In the task inventory, the designer identifies the steps, or task elements and sub-elements, using observation and interview techniques. Analysis of tasks qualifies the task description and analyzes the behavior using learning hierarchy analysis (Gagne, 1965, 1977, 1985) or Bloom's taxonomy (Bloom, Krathwohl, Masia, 1956). No explicit technique is described for sequencing tasks, except those implied by the task analysis step. Hershbach essentially applies classic task analysis procedures to industrial education.