THE EFFECTS OF STRUCTURED DEVELOPMENT METHODS ON THE JOB SATISFACTION OF PROGRAMMER/ANALYSTS:

A THEORETICAL MODEL

David K. Goldstein

May 1982

Sloan WP # 1330 - 82

CISR WP # 90

Center for Information Systems Research

Massachusetts Institute of Technology
Sloan School of Management
77 Massachusetts Avenue
Cambridge, Massachusetts, 02139

THE EFFECTS OF STRUCTURED DEVELOPMENT METHODS ON THE JOB SATISFACTION OF PROGRAMMER/ANALYSTS:

A THEORETICAL MODEL

David K. Goldstein

May 1982

Sloan WP # 1330 - 82

CISR WP # 90

Center for Information Systems Research

Sloan School of Management

Massachusetts Institute of Technology

ABSTRACT

This paper presents a theoretical model that explains the effects of the use of structured development methods on the job satisfaction of programmer/analysts. In the model, the independent variable - the use of structured methods - is hypothesized to have an overall positive effect on the principle dependent variable - the job satisfaction of programmer/analysts. This can be best explained through several intervening variables - role conflict, role ambiguity, task achievement, skill variety, task identity, and autonomy. These are variables that will be significantly affected by the introduction of structured methods and will significantly affect the job satisfaction of programmer/analysts. This paper describes the theoretical model and outlines a quasi-experiment that can be used to test the model.

Over the past ten years, the use of various types of structured methods, from structured programming to structured systems analysis, has been advocated as a way of improving productivity and quality in systems development. Proponents of structured methods claim that they simplify systems development by providing guidelines for carrying out the phases of the development lifecycle and by providing a language that facilitates the description and communication of systems requirements and systems designs. This allows programmer/analysts to develop systems more quickly; it also reduces systems errors, and leads to the development of systems that better meet user needs (Goldstein, 1982a).

One potential consequence of the use of structured methods - its effect on the job satisfaction of programmer/analysts - has received very little attention. Some researchers have argued that the use of structured methods reduces the skill level of programmer/analysts (Kraft, 1977), which could lead to a decrease in job satisfaction. However, we could argue that the use of structured methods reduces conflict between programmer/analysts and users, reduces some of the ambiguity in systems development, and leads to increased job satisfaction.

Level of job satisfaction could have important consequences for MIS managers. In many studies job satisfaction has been shown to be negatively related to absenteeism and turnover (Locke, 1976). Turnover is of special importance in MIS, due to the shortage of programmer/analysts and the high cost of training new programmer/analysts. If the use of structured methods decreases job satisfaction, then its personnel costs could outweigh its benefits. Alternatively, if job satisfaction increases with the use of structured methods, this would provide further evidence for those advocating its use.

This paper presents a model that explains the effects of the use of structured methods on the job satisfaction of programmer/analysts. In the model, the independent variable - the use of structured methods - is hypothesized to have an overall positive effect on the principle dependent variable - the job satisfaction of programmer/analysts. This can be best explained through several intervening variables - variables that will be significantly affected by the introduction of structured methods and will significantly affect the job satisfaction of programmer/analysts. The model also includes a set of environmental variables - factors that could affect the validity of the model.

This paper is divided into four sections. The first section reviews the relevant research on structured methods and job satisfaction. The second discusses the theoretical model. The third outlines a quasi-experimental design that can be used to test the model. The fourth discusses the significance of the research.

REVIEW OF RELEVANT LITERATURE

There is no research that directly examines the effects of structured development methods on the job satisfaction of programmer/analysts. There is, however, a large body of literature that discusses the impacts of structured methods on the systems development process and an even larger body of literature that discusses the determinants of job satisfaction. There are also a few articles that examine the determinants of job satisfaction in programmer/analysts.

Impacts of Structured Methods on Systems Development

Many authors have described how structured methods are used in systems development. Various types of structured methods, such as HIPO (Jones, 1976), structured design (Stevens, Myers, and Constantine, 1974), and data structure design (Jackson, 1975; Warnier, 1974) have been used to aid programmer/analysts in systems analysis, design, and programming.

Goldstein (1981), Mendes (1980), Canning (1979a), Jones (1976), and Winters (1979) describe how different structured methods are used in systems analysis. Programmer/analysts use these methods to model both the functions performed and data used by a business, as well as what the proposed systems solution will do. In the articles, the authors claim that programmer/analysts using structured methods in systems analysis develop a better understanding of the business problem. They are better able to communicate their understanding of the business problem and the proposed systems solution to systems users. These two factors lead to a reduction in systems analysis errors caused by analysts misunderstanding the users' business problem or the users misunderstanding the analysts' proposed solution. This should lead to the development of systems that better meet the requirements of users.

Canning (1979b), Bernstein (1972), Menard (1978), and Hamilton and Block (1979) discuss the use of structured methods in the systems design phase. In systems design, structured methods are used to describe how a system will function. They can describe the modules the system will use, the interfaces between modules, and the data structures that must be developed. The authors claim that the design aids are used to structure and simplify the programming process. The output of the design process is easier to turn into programs when these aids are used. This allows

organizations to use less experienced programmers or to increase the productivity of their more experienced programmers. The number of programming errors are reduced, maintenance is simplified, and project management is easier when structured methods are used to support systems design.

Canning (1974a,b), Baker (1972), Inmon (1976), and Rader (1978) describe the impact of structured methods on the programming phase. In many cases design and programming methods are used together. The design methods produce modules that are programmed using structured programming methods. The programming methods are used to structure the programming process. The benefits of these methods in programming are the same as they are in design.

In some related research, Kraft (1977) and Greenbaum (1979) claim that the use of structured methods, such as structured programming and Chief Programmer Teams, de-skill the programming task. This makes it easier for programming to be carried out by less experienced and less expensive personnel. They draw analogies between the effect of structured methods on programmer/analysts and the effect of the assembly line and scientific management techniques on factory workers.

Determinants of Job Satisfaction

There is a great deal of research on job satisfaction and its determinants. Locke (1976) provides a summary of the research on the determinants of job satisfaction. He divides the determinants into events and conditions, and agents.

Among events and conditions, he describes the effects on job satisfaction of the job itself, the pay, promotion, recognition, and working conditions. Hackman and Oldham (1980) identify several characteristics of

the job itself that positively affect job satisfaction. They are:

> Skill variety - The degree to which a job requires a variety of different activities in carrying out the work, involving a number of different skills and talents of the person.
>
> Task identity - The degree to which a job requires completion of a whole and identifiable piece of work, that is, doing a job from beginning to end with a visible outcome.
>
> Task significance - The degree to which the job has a substantial impact on the lives of other people, whether those people are in the immediate organization or in the world at large.
>
> Autonomy - The degree to which the job provides substantial freedom, independence, and discretion to the individual in scheduling the work and in determining the procedures to be used in carrying it out.
>
> Job feedback - The degree to which carrying out the work activity required by the job provides the individual with direct and clear information about the effectiveness of his or her performance (Hackman and Oldham, 1980, pages 78-79).

Hackman and Oldham found significant positive correlations between each of these job characteristics and job satisfaction.

One aspect of work not considered by Hackman and Oldham is the amount of task achievement found in the job. Locke claims that the degree to which workers can overcome the challenges of their jobs positively affects their job satisfaction. He also stresses that individual differences can moderate the effects of these factors on job satisfaction.

Among agents, Locke describes the effects of self-perception, co-workers, and the organization on job satisfaction. Inasmuch as the use of structured methods changes the job performed by programmer/analysts, it can change their self-perception. The literature on the impacts of structured methods suggests that the use of structured methods could also affect the relationship of programmer/analysts to users and managers. Role ambiguity and role conflict (Kahn, et al., 1964) are two constructs that measure the degree of ambiguity in a job and the degree to which a worker is subject to conflicting demands from co-workers and managers. Both

constructs have been shown to be negatively related to job satisfaction. This is especially true for boundary spanning jobs - jobs requiring a great deal of intra- or interorganizational contact.

Job Satisfaction of Programmer/Analysts

A few researchers have attempted to examine the determinants of job satisfaction in programmer/analysts. Awad (1977) and Willoughby (1972) used a needs reinforcement model to examine the factors affecting job satisfaction of programmer/analysts. Willoughby found satisfaction was highest when high levels of ability utilization, achievement, advancement, creativity, recognition, responsibility, company fairness, social status, and supervisor fairness were present. Awad applied the needs reinforcement model to both programmers and analysts in one company. He found no differences in needs between programmers and analysts. He also found that a significant correlation between needs-reinforcement correspondence and job satisfaction.

Bostrom (1980) examined the effects of role conflict and role ambiguity on job satisfaction of system designers. He considered the effects of these variables on 75 user-designer pairs involved in systems maintenance. He found that role conflict and role ambiguity were significantly negatively correlated with job satisfaction.

Couger and Zawacki (1981) used Hackman and Oldham's model to examine the effects of job characteristics on job satisfaction of programmer/analysts. They conducted a large sample survey with 1000 programmers and analysts at many companies and government agencies. They found that characteristics of the job, as measured by the job's motivating potential, correlated positively with job satisfaction.

The job satisfaction literature identifies several factors that are determinants of satisfaction in programmer/analysts including characteristics of their job and characteristics of their relationship with others in the organization. The structured methods literature provides insight into how the methods are used and how they might affect the jobs and the interactions of programmer/analysts. In the next section, these two areas will be drawn on to develop a model of how the use of structured methods affects the job satisfaction of programmer/analysts.

THEORETICAL MODEL

Figures 1 and 2 present the theoretical model that relates the use of structured methods to job satisfaction. In Figure 1, the independent variable - the use of structured methods - is hypothesized to have an overall positive effect on the principal dependent variable - the job satisfaction of programmer/analysts. This effect can be best explained by examining the effects of the independent variable on a set of intervening variables - role conflict, role ambiguity, task achievement, skill variety, task identity, and autonomy - which have been shown to be related to job satisfaction in other studies. This research will show that these variables are significantly affected by the introduction of structured methods and that they are significantly related to job satisfaction in programmer/analysts. Figure 2 relates the independent variable to two performance variables - the productivity of programmer/analysts and the quality of their work. It is hypothesized that the use of structured methods will positively affect these performance variables, but that this will have no significant effect on job satisfaction. This section describes

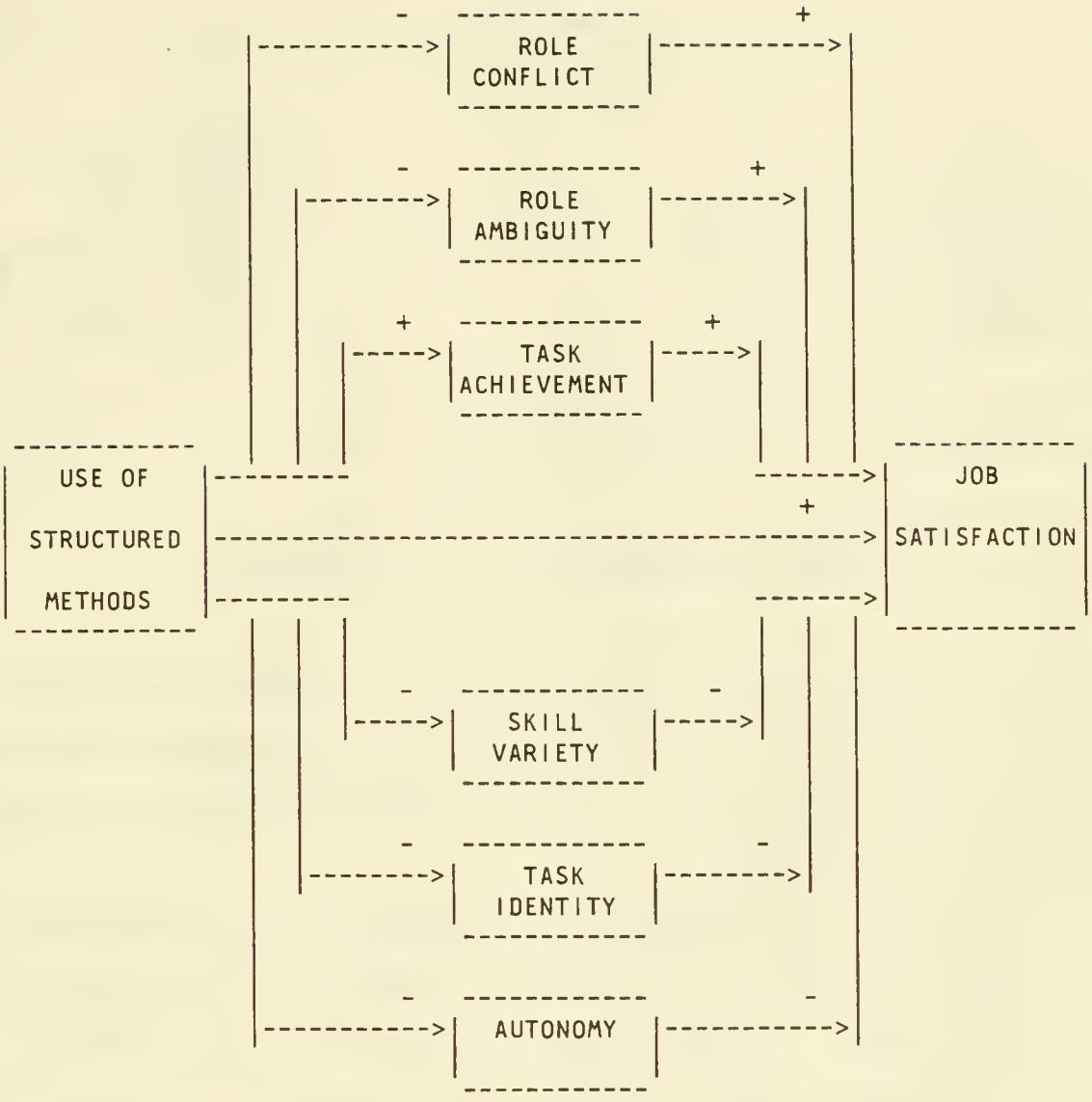

FIGURE 1: HYPOTHESIZED EFFECTS OF STRUCTURED METHODS ON JOB CHARACTERISTICS AND JOB SATISFACTION

the dependent variable, the independent and intervening variables, the research hypotheses, and the environmental variables. The environmental variables measure factors that could affect the validity of the model.

Dependent Variables

Job satisfaction of programmer/analysts is the main dependent variable in the model. Locke defines job satisfaction as "a pleasurable or positive

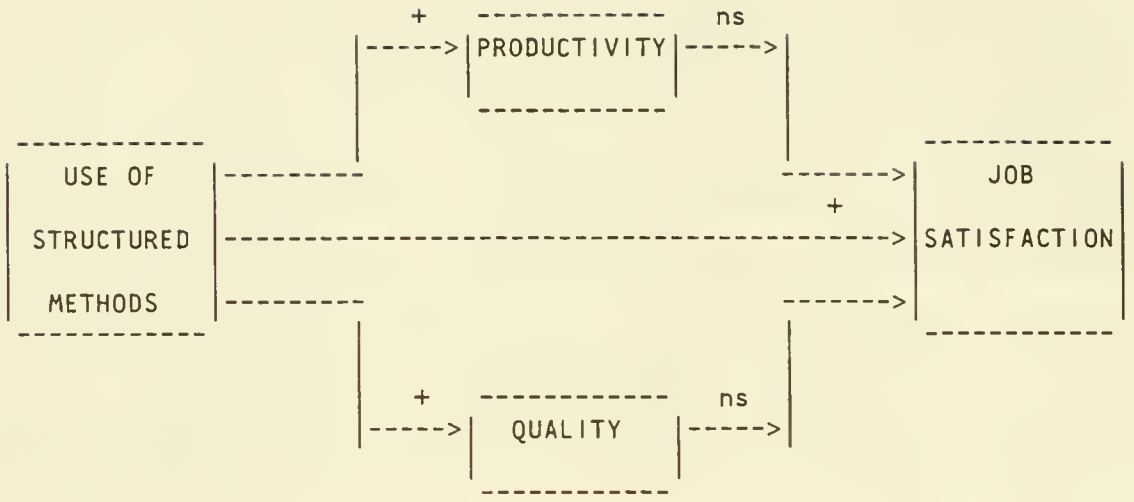

FIGURE 2: HYPOTHESIZED EFFECTS OF STRUCTURED METHODS ON PERFORMANCE AND JOB SATISFACTION

emotional state resulting from the appraisal of one's job or job experience (Locke, 1976, page 1300)." In the job satisfaction literature, it has been related strongly (negatively) to turnover and absenteeism and less strongly (positively) to performance.

Although job satisfaction will be the focus of our investigation, it will also be valuable to examine the effects of the introduction of structured methods on other dependent variables. The literature on structured methods claims that their use improves the performance of programmer/analysts. There are two main aspects of performance that can be examined - the effects of structured methods on the quality of work done by programmer/analysts and their effects on the productivity of programmer/analysts.

These dependent variables will not be emphasized in our study for two reasons. First, there is little dispute that using structured methods improves the quality of systems and that it improves the productivity of programmer/analysts. Second, there is a great deal of dispute about how to measure program quality and programmer productivity. This will negatively

affect the credibility of any results obtained in this area.

Independent Variable

The use of structured methods is the independent variable in the model. We will define structured methods as a set of procedures that describe how to develop business application systems. They include a language for representing how a system functions and a set of guidelines for using the language for systems analysis, design and programming. The language is used to describe how the business functions before the introduction of a computer system, how the business will function after the system is developed, and how the system itself functions. It is generally made up of a set of diagrams that describe functional and data hierarchies, and data flows.

The guidelines describe how to get information about the business from the system user, how to describe to the user what the system will do, and how to describe the modules and data structures that make up the system. In some cases, the methods are supported by automated tools that aid in the generation of the diagrams, check the consistency of the diagrams, and generate code from the lowest level of the diagrams.

There are several examples of structured methods that have been advocated by various groups. One example is the data structure approach developed by Michael Jackson (1975). Data structure diagrams are used to describe the data being used by the business and to describe the business' data needs. The system structure is built around the data structure. Jackson's method as well as three other development methods are described in a recent article by Bergland (1981). Exxon has enhanced the Jackson approach and added more specific guidelines and automated aids. They call the analysis method SSA and the design and programming method PST. Other

examples include the PRIDE/ASDM method described by Canning (1981), which consists of a nine phase development methodology supported by automated aids, Softec's SADT (Ross, 1977), and the group of structured methods developed and marketed by Yourdon (1975).

Automated programming aids (e.g., screen formatters), programming methods (e.g., structured programming), and analysis methods (e.g., information analysis), would not be included in the above definition of structured methods. These development aids do not meet the requirements described above.

There are several reasons for choosing the use of structured methods, as defined above, as the independent variable. First, structured methods are currently used by many organizations. This facilitates the selection of a site and also makes the results relevant to many people. Second, they have a large effect on the jobs performed by programmer/analysts. They cause changes in the analysis, design, and coding phases. This should make it easier to detect changes in job satisfaction. Third, they are controversial. Based on the research of Kraft (1977) and Greenbaum (1979), we could argue that the use of structured methods de-skills programmer/analysts and reduces their job satisfaction. Based on the cases discussed above, we could argue that the use of structured methods facilitates systems development, reduces role conflict and ambiguity, and leads to an increase in job satisfaction.

Intervening Variables

The literature on job satisfaction identifies a number of variables as determinants of job satisfaction. Some of these variables should also be affected by the introduction of structured methods. We will call them

intervening variables. They allow us to better understand how the use of structured methods leads to changes in job satisfaction. The six variables are described below.

Role conflict - Role conflict is "the degree of incongruity or incompatibility in the expectations or requirements communicated to a focal person (Bostrom, 1980, page 92)." In jobs that require workers to deal with many departments within an organization or that require dealing with people outside the organization, role conflict is common. The task of systems development requires a great deal of contact with people outside of the project team. These include users at all levels, d. p. management, technical support staff, and operations staff.

Kahn et al. (1964) have identified several components of role conflict. They are:

> Person-role Conflict - the extent to which role expectations are incongruent with the orientations, standards, or values of the focal person.
>
> Intrasender Conflict - the extent to which role requirements are incompatible with the resources or capabilities of the focal person.
>
> Intersender Conflict - the extent to which role requirements or expectations from one party oppose those from one or more other parties.
>
> Role Overload - the extent to which the various role expectations communicated to the focal person exceed the amount of time available for their accomplishment (Bostrom, 1980, page 93).

Bostrom examined the effects of different components of role conflict on the job satisfaction of programmer/analysts. He found a significant negative relationship between person-role, intrasender, and intersender role conflict and job satisfaction. Since structured methods should facilitate communication between programmer/analysts and other members of the organization, they should reduce the level of role conflict experienced by the programmer/analysts.

Role ambiguity - Role ambiguity is "the degree to which desired expectations are vague, ambiguous, or unclear, thereby making it difficult for the person to fulfill the requirements [of his role] (Bostrom, 1980, page 93)." Bostrom found the level of role ambiguity perceived by programmer/analysts was significantly negatively related to their level of job satisfaction. Since using structured methods should clarify the tasks to be performed by programmer/analysts, it should reduce the level of role ambiguity that they experience.

Task achievement - Researchers have found that the degree to which workers can overcome the challenge of their job positively affects their job satisfaction (Locke, 1976). Since structured methods simplify systems development, programmer/analysts should perceive an increase in task achievement.

Three of the variables identified by Hackman and Oldham (1980) - skill variety, task identity, and autonomy - should be affected by the use of structured methods. By limiting the tasks performed by programmer/analysts, the use of structured methods should reduce skill variety. Further, it should reduce task identity and autonomy by encouraging the division of development projects into relatively independent modules.

Hypotheses

The hypotheses relate 1) the use of structured methods to job satisfaction through the intervening variables described above and 2) the use of structured methods to productivity and quality in systems development.

H1: The use of structured methods will cause an overall increase in the job satisfaction of programmer/analysts. It will simplify and structure

the tasks performed by programmer/analysts and it will improve communication between programmer/analysts and users. Thus, system developers will feel that their job is more manageable and react more favorably to it. We can better explain the positive effects of the use of structured methods on job satisfaction by examining their effects on some of the intervening variables discussed above.

H2(A): The use of structured methods will decrease role conflict perceived by programmer/analysts. It will improve communication between programmer/analysts and others involved in systems development.

H2(B): Role conflict will be negatively related to job satisfaction.

H3(A): The use of structured methods will decrease role ambiguity perceived by programmer/analysts. It will lead to a better specification of the tasks to be performed by programmer/analysts.

H3(B): Role ambiguity will be negatively related to job satisfaction.

H4(A): The task achievement of programmer/analysts will increase with the use of structured methods. The structured methods will make the task of systems development easier and hence programmer/analysts will feel that they can better do their jobs.

H4(B): Task achievement will be positively related to job satisfaction.

Some of the job characteristics defined by Hackman and Oldham will be negatively affected by the introduction of structured methods. This will moderate the positive effects of structured methods on job satisfaction. Specifically, three job characteristics will be negatively affected.

H5(A): The use of structured methods will reduce the skill variety perceived by programmer/analysts. It will limit the options available to the programmer/analyst in systems development.

H5(B): Skill variety will be positively related to job satisfaction.

H6(A): The use of structured methods will reduce the task identity perceived by programmer/analysts. The development task will be split into smaller tasks when structured methods are used.

H6(B): Task identity will be positively related to job satisfaction.

H7(A): The use of structured methods will limit the level of autonomy perceived by programmer/analysts. Using structured methods will allow project managers to better specify the tasks to be performed by programmer/analysts.

H7(B): Autonomy will be positively related to job satisfaction.

H8(A): The introduction of structured methods will improve the productivity of programmer/analysts and it will improve the quality of their work. This is based on the previous research on the impact of structured methods on productivity and quality.

H8(B): Changes in productivity and quality will not have a significant effect on job satisfaction. The research with other types of jobs has shown little relationship between job satisfaction and performance (Locke, 1976). There is some evidence that there is a relationship between current performance and future satisfaction, but not between current job satisfaction and current performance (Wanous, 1974). Locke claims that individual differences, measurement problems, and contextual differences play a major role in moderating the relationship between performance and job satisfaction. Controlling for these factors is out of the scope of this study.

Environmental Variables

The hypotheses discussed above can be tested with a quasi-experiment. That is, we can examine differences in the dependent variable and in the intervening variables between a treatment group - a group of programmer/analysts using structured methods - and a control group - a group of programmer/analysts not using structured methods. However, in a quasi-experiment we cannot control for differences between the treatment and control group or for peculiarities of a particular research site. We can attempt to measure as many of these factors, which we will call environmental variables, as possible. This will allow us to determine the degree of validity of the quasi-experiment. In this section, we will consider three types of environmental variables - characteristics of individual programmer/analysts, characteristics of project teams, and characteristics of the research site.

We will assume the quasi-experiment is a pretest-posttest with control group design. That is, we will be concerned with changes in job satisfaction between the pretest and posttest and with differences in the amount of the changes between the treatment and control groups.

The environmental variables can threaten both the internal and external validity of the quasi-experiment. For example, differences between the programmer/analysts in the treatment and control groups could falsely lead us to attribute a change in job satisfaction to the introduction of structured methods - a threat to internal validity. In addition, if there were significant differences between the programmer/analysts at the research site and programmer/analysts at other organizations, the generalizability of the quasi-experiment would be affected - a threat to external validity. In this section we will rely on Cook and Campbell's (1979) classification of

threats to validity.

Characteristics of programmer/analysts - Differences in several characteristics of programmer/analysts in the treatment and control groups could affect both internal and external validity. This includes background variables, such as job longevity, age, amount of systems development experience, tenure in the project group, amount of experience with structured methods, and level of motivation and satisfaction with job contexts.

For example, differences in job longevity could threaten the internal validity of the quasi-experiment. Suppose the treatment group contained a much larger percentage of new hires than the control group. We would expect that the job satisfaction of the new hires would decline after six months when the novelty of their new job wears off. This change in job satisfaction would be independent of the introduction of structured methods. We could, however, falsely attribute this change to the introduction of structured methods. Cook and Campbell would classify this as a threat to internal validity due to the interaction of selection and maturation.

Background differences between the treatment and control groups could also threaten external validity. For example, if the treatment group was older or more experienced in systems development or more experienced with structured methods than the control group, we could not be sure that the results of the quasi-experiment could be generalized to other treatment groups containing less experienced programmer/analysts. Further, Katz (1977) has shown that age and job longevity differences moderate the effects of job characteristics on job satisfaction. If most of the members of the treatment group were working at the company for at least ten years, it would be difficult to generalize the results to organizations containing less

experienced programmer/analysts. Cook and Campbell would classify this as a threat due to the interaction of selection and treatment.

Other individual differences, such as level of motivation and satisfaction with job contexts, could also affect the generalizability of the quasi-experiment. Hackman and Oldham (1980) identify these as constructs that moderate the effects of job characteristics on job satisfaction. Suppose the treatment group was significantly more motivated than the control group. We would then be uncertain that any differences in the level of job satisfaction between the groups were due to the use of structured methods or to the interaction of structured methods with motivation.

Project team characteristics - Differences in project teams could affect external validity. One aspect of team differences is the leadership style of the project leader and of the programmer/analyst's peers. Bowers and Seashore (1966) have identified four leadership characteristics:

> Support - behavior that enhances someone else's feeling of personal worth and importance.
>
> Interaction Facilitation - behavior that encourages members of the group to develop close, mutually satisfying relationships.
>
> Goal Emphasis - behavior that stimulates an enthusiasm for meeting the groups goal or achieving excellent performance.
>
> Work Facilitation - behavior that helps achieve goal attainment by such activities as scheduling, coordinating, planning, and by providing resources such as tools, materials, and technical knowledge (Bowers and Seashore, page 247).

These characteristics can be applied to both the project leaders and other members of the project team. Bowers and Seashore found significant positive correlations between measures of these characteristics and job satisfaction and performance. Yunger and Hunt (1976) found these characteristics similar to the characteristics identified in the Ohio State LBDQ leadership scales.

In addition, differences in the background of the project leaders and differences in the amount of time the team has worked together - team longevity - could affect could affect external validity. Katz (1979) has found that group longevity had a significant effect on performance in research and development groups.

Project team differences between the treatment and control groups would have the same effect on external validity as individual differences. We would be uncertain whether to attribute changes in the dependent variable to the independent variable or, to the interaction of the independent variable and the project team differences.

Research site characteristics - The characteristics of the research site could affect both internal and external validity. For example, the target organization could select only the most satisfied programmer/analysts to be used in the treatment group. If this were the case, we would expect the satisfaction of this group to regress to the mean between the pretest and the posttest - independent of the introduction of structured methods. We could then falsely attribute their change in job satisfaction to the use of structured methods. Cook and Campbell call this a threat to internal validity due to statistical regression.

Differences in how structured methods were used at the research site could also threaten the validity of the quasi-experiment. If the programmer/analysts in the control group determined that structured methods were a valuable tool, they could start imitating the treatment group by using some of the structured techniques. The differences in job satisfaction between the treatment and control group could then be moderated by the imitation of the treatment. A similar problem would occur if the treatment group did not become proficient in the use of structured methods

between the pretest and the posttest.

Organizational characteristics of the research site could also affect external validity. For example, if the d. p. organization has had a number of changes in top management in the last year, if they have introduced a new compensation scheme, or if they have had a large number of new hires or layoffs this could affect programmer/analyst's reaction to the introduction of structured methods. This would limit the generalizability of the quasi-experiment to other organizations.

QUASI-EXPERIMENTAL DESIGN

The model described above can be tested using a pretest-posttest with control group design. Cook and Campbell (1979) diagram this design as:

$$\begin{array}{ccc} O & X & O \\ O & & O \end{array}$$

The design can be implemented by finding one company that is about to introduce structured methods to some, but not all, of its programmer/analysts. We can then administer a questionnaire to all the programmer/analysts in the company right before structured methods are introduced to measure the independent, intervening, and environmental variables (the pretest). The questionnaire can then be readministered six months later (the posttest). The group of programmer/analysts who do not use structured methods can serve as a control group at the pretest and posttest. This section outlines the quasi-experimental design. It discusses the requirements of the research site, the experimental procedure, the measures, and the method of analysis. Goldstein (1982b) provides a more detailed discussion of the design.

Research Site - The research site should be a large organization that is about to introduce structured methods on some of its project groups. Ideally, the organization should have 100 or more programmer/analysts. It should develop mostly large transaction processing and management information systems that span boundaries within the organization. These are the types of systems most suited for development with structured methods. The organization should not have suffered any recent organizational disruptions, such as the ones discussed in the previous section, that could overshadow the effects of the introduction of structured methods. We should be able find the organization through a company involved in marketing structured methods.

Experimental procedure - The questionnaire should be administered to all the programmer/analysts in the organization and their project leaders right before the introduction of structured methods and six months after their introduction. The structured methods should be used by at least 30 programmer/analysts.

Measures - Measures have already been developed for many of the variables in the model. For example, the scales developed by Rizzo, et al. (1970) and modified by Bostrom (1980) can be used to measure role conflict and ambiguity. Hackman and Oldham's (1980) Job Diagnostic Survey contains scales that measure skill variety, task identity, and autonomy. The JDS also contains items to measure job satisfaction, level of motivation, and satisfaction with job contexts. Bowers and Seashore's (1966) measure can be used to measure the leadership characteristics of each programmer/analysts' peers and project leader. Other items can be added to the questionnaire to measure task achievement, the other characteristics of programmer/analysts identified above, and the programmer/analysts'

proficiency with structured methods.

Project leaders should be given a separate questionnaire. It should be used to obtain information on productivity and quality for their programmer/analysts, to obtain information on their team and on their own background, and to verify the intervening variable measures obtained from their programmer/analysts.

Method of Analysis - Simple gain score analysis can be used to test the hypotheses. This is an analysis of variance technique that examines the differences in changes in a variable from the pretest to the posttest. The assumption is that the treatment will lead to more (or less) change in the treatment group than in the control group (Cook and Campbell, 1979).

The characteristics of the treatment and control groups can be examined to determine the validity of our results. Any significant differences between the treatment and control groups on the background, individual differences, and team differences measures could affect the internal validity of our results. Further, the characteristics of the entire population will give us some idea of the generalizability of our results.

DISCUSSION

This research will provide some insights into the effects of the use of structured methods on the job satisfaction of programmer/analysts. Using a pretest-posttest with control group design allows us to isolate the effects of structured methods. The intervening variables provide further explanation as to how the use of structured methods affects job satisfaction.

There are, however, some potential problems with the study. First, the six month time frame might not be long enough for programmer/analysts to become familiar with structured methods. However, a longer time frame would create problems with experimental mortality. Turnover would cause people to drop out of the experiment between the pretest and the posttest.

Second, differences between treatment and control groups and between the experimental groups and the population of programmer/analysts could affect the generalizability of the results. Although this is hard to control for in a quasi-experimental design, the presence of multiple measures of individual and team differences should point out any potential problems.

A larger sample survey could eliminate some of the problems. The instruments developed in this quasi-experiment could be used to compare the programmer/analysts that use structured methods to those that do not use structured methods in several companies that have partially implemented structured methods. If the results of the pretest-posttest quasi-experiment were confirmed with this posttest only design, this would provide stronger evidence that the hypotheses were true.

Finally, this research should lead to further research in two areas. First, this research will provide some insights into the determinants of job satisfaction in programmer/analysts. Further research could explore the effects of the factors identified here, as well as other factors, on job satisfaction. Research could also explore the effects job enrichment programs or other work redesign efforts on programmer/analysts. This is an important research, because of the relationship between job satisfaction and job outcomes such as productivity and turnover. These are critical problems in d. p. organizations.

Second, the research provides some insights into the effects of job changes on job satisfaction. The introduction of new office technologies and the more widespread use of computers will change the jobs of many workers. Further research could examine the effects of these new technologies on job satisfaction. This would give us a better idea of the possible costs and benefits of office automation or other technologies.

REFERENCES

Awad, Elias M., "Prediction of Satisfaction of Programmers, Analysts," Data Management, January, 1977, 12-18.

Baker, F. T., "Programmer Team Management of Production Programming," IBM Systems Journal, Volume 11, Number 1, 1972, pages 56-75.

Bergland, G. D., "A Guided Tour of Program Design Methods," Computer, October, 1981, pages 13-37.

Bernstein, C. M., "Experience with Exxon's Implementation of the Jackson Program Design Method," National Computer Conference Proceedings, 1978, pages 636-637.

Bostrom, Robert, P., "Role Conflict and Ambiguity: Critical Variables in the User-Designer Relationship," Proceedings of the Seventeenth Annual Computer Personnel Research Conference, 1981, pages 88-112.

Bowers, David G. and Stanley E. Seashore, "Predicting Organizational Effectiveness with a Four-Factor Theory of Leadership," Administrative Science Quarterly, 1966,pages 238-263.

Canning, R. G., Editor, "The Search for Software Reliability," EDP Analyzer, Volume 12, Number 5, May, 1974a.

Canning, R. G., Editor, "The Advent of Structured Programming," EDP Analyzer, Volume 12, Number 6, June, 1974b.

Canning, R. G., Editor, "The Analysis of User Needs," EDP Analyzer, Volume 17, Number 1, January, 1979a.

Canning, R. G., Editor, "The Production of Better Software," EDP Analyzer, Volume 17, Number 2, February, 1979b.

Canning, R.G., Editor, "Application System Design Aids," EDP Analyzer, Volume 19, Number 10, October, 1981.

Cook, Thomas D. and Donald T. Campbell, Quasi-Experimentation Design and Analysis Issues for Field Settings, Houghton Mifflin Company, Boston, 1979.

Couger, Daniel and Robert Zawacki, Motivating and Managing Computer Personnel, John Wiley and Sons, New York, 1981.

Goldstein, David K., "Exxon's Use of Structured Systems Analysis: A Case Study," Unpublished paper, 1981.

Goldstein, David K., "The Impact of Development Aids on the Systems Development Process," Proceedings of the Conference on Human Factors in Computer Systems, 1982a, pages 130-134.

Goldstein, David K., "The Effects of Structured Development Methods on the Job Satisfaction of Programmer/Analysts: A Quasi-Experimental Design," Unpublished paper, 1982b.

Greenbaum, Joan M., In the Name Of Efficiency, Temple University Press, Philadelphia, 1979.

Hackman, J. R. and G. R. Oldham, Work Redesign, Addison Wesley, Reading, Mass., 1980.

Hamilton, Kenneth and Arthur Block, "Programmer Productivity in a Structured Environment," Infosystems, April, 1979, pages 44-50 and May, 1979, pages 62-68.

Inmon, Bill, "An Example of Structured Design," Datamation, March, 1976, pages 82-86.

Jackson, Michael A., Principles of Program Design, Academic Press, 1975.

Jones, Martha N., "HIPO for Developing Specifications," Datamation, March, 1976, pages 112-125.

Kahn, R. L. , Wolfe D., Quinn R., Snoek J. D., and Rosenthal R., Organizational Stress: Studies in Role Conflict and Role Ambiguity, John Wiley, New York, 1964.

Katz, Ralph, "Satisfaction and Behavioral Reactions of Employees to Task Characteristics as a Function of Job Longevity, Age, and High-Order Needs," Sloan Working Paper, 932-77, MIT, Cambridge, Mass, 1977.

Katz, Ralph, "The Influence of Group Longevity on Project Communication and Project Performance," Sloan Working Paper, 1179-80, MIT, Cambridge, Mass, 1980.

Kraft, Philip, Programmers and Managers, Springer-Verlag, New York, 1977.

Locke, E. A., "The Nature and Causes of Job Satisfaction," in Dunnette, M. D., Ed., Handbook of Industrial and Organizational Psychology, Rand McNally, 1976.

Menard, Jayne B., "Exxon's Experience with the Michael Jackson Design Method," Data Base, Volume 11, Number 3, Winter-Spring, 1980, pages 88-92.

Mendes, Kathleen S., "Structured Systems Analysis: A Technique to Define Busines Requirments," Sloan Management Review, Volume 24, Number 4, Summer, 1980, pages 51-63.

Rader, J. A., "Experience with an Application of Structured Design," National Computer Conference Proceedings, 1978, pages 633-636.

Rizzo, J. R., R. J. House, and S. J. Lirtzman, "Role Conflict and Ambiguity in Complex Organizations," Administrative Science Quarterly, Volume 15, 1970, pages 150-163.

Ross, Douglas T., "Structured Analysis (SA): A Language for Communicating Ideas," IEEE Transactions on Software Engineering, Volume SE-3, Number 1, January, 1977, pages 16-21.

Stevens, W. P., G. J. Myers, and L. L. Constantine, "Structured Design," IBM Systems Journal, Volume 13, Number 2, 1974, pages 115-139.

Wanous, J. P., "A Causal-Correlational Analysis of the Job Satisfaction and Performance Relationship," Journal of Applied Psychology, Volume 59, 1974, pages 139-144.

Warnier, J. D., Logical Construction of Programs, Van Nostrand, Reinhold, New York, 1974.

Willoughby, T. C., "Staffing the MIS Function," ACM Computing Surveys, Volume 4, Number 4, 1972, pages 241-259.

Winters, Edward W., "Experience with Problem Statement Language: A Tool for Structured Documentation," Application Development Symposium Proceedings, 1979, pages 137-141.

Yourdon, E., and L. L. Constantine, Structured Design, Yourdon Press, New York, 1975.

Yunger, G. W. and J. G. Hunt, "An Empirical Comparison of the Michigan Four-Factor and the Ohio State LBDQ leadership Scales," Organizational Behavior and Human Performance, Volume 17, 1976, pages 45-65.